Distributed Solutions for Energy Efficiency Fairness in Multicell MISO Downlink

Kien-Giang Nguyen, *Student Member, IEEE*, Quang-Doanh Vu, Markku Juntti, *Senior Member, IEEE*, and Le-Nam Tran, *Senior Member, IEEE*

***Abstract*—This paper aims at guaranteeing the achievable energy efficiency (EE) fairness in a multicell multiuser multiple-input single-output downlink system. The design objective is to maximize the minimum EE among all base stations (BSs) subject to per-BS power constraints. This results in a max-min fractional program and as such is difficult to solve in general. Our goal is to develop decentralized algorithms for the max-min EE problem based on combining successive convex approximation (SCA) framework and alternating direction method of multipliers (ADMM). Specifically, leveraging the SCA principle, we iteratively approximate the nonconvex design problem by a sequence of convex programs for which two decentralized algorithms are then proposed. In the first approach, the convex program obtained at each step of the SCA procedure is solved optimally by allowing the BSs to exchange the required information until the ADMM converges. The convergence of the first method is analytically guaranteed but the amount of backhaul signaling can be noticeable in some realistic settings. To reduce the backhaul overhead, the second method performs an abstract version of the ADMM where only one variables update is carried out. Numerical results are provided to demonstrate the effectiveness of the two proposed decentralized algorithms.**

***Index Terms*—Energy efficiency, max-min fractional programming, successive convex approximation, alternating direction method of multipliers.**

I. Introduction

Multiple-antenna techniques can offer impressive improvements on the achievable capacity of wireless communications systems. Previously, a major objective in network design has been to maximize the *spectral efficiency* (SE) to satisfy the increasing demand on data traffic of cellular networks. While the SE maximization problem still remains important, we also need to consider the total energy spent by the wireless systems, especially due to the recent and on-going explosive growth of the number of wireless devices and data traffic volumes. As a result, energy-efficient transmission approaches have been the main focus in a large portion of recent works (cf. [2] for a relatively comprehensive survey).

Energy efficiency (EE), also known as *bit-per-Joule capacity*, is defined as the ratio of throughput and the total power consumption of the network [2]. The problem of EE maximization under individual quality-of-service constraints and/or transmit power constraints has been of particular interest [3]–[9]. To maintain fairness of the individual cells or base stations (BSs), maximizing the minimum of EEs of users was considered [10]–[12]. This is relevant, e.g., in cellular networks where BSs are not connected to fixed electricity grid. The problem leads to solving a max-min fractional program, which is nonconvex and generally difficult to solve to global optimality.

For nonconvex problems in general, and for the max-min EE problem in particular, a classical goal is to find a stationary solution, i.e., a solution that satisfies the Karush-Kuhn-Tucker (KKT) conditions. This was done in [10] by combining the Dinkelbach's approach and alternating optimization. More explicitly, the parameterized problem attained from the Dinkelbach's method is solved by alternately optimizing the beamformers, receivers and other auxiliary variables. To improve this approach, in [13], we proposed a more computationally efficient beamforming design for the max-min EE problem, using an inner approximation framework which is now probably better known as the successive convex approximation (SCA). Therein, novel transformations were introduced to expose the hidden convexity of the considered problem, and the remaining nonconvexity is iteratively replaced by convex approximations. In [12], the SCA principle together with fractional programming was also used to derive power control policies for the max-min EE problem for massive MISO systems. Furthermore, a general framework combining the SCA approach and fractional programming for energy-efficient resource allocation problems was presented in [14].

The existing beamforming designs or power control policies for the max-min EE problem are centralized algorithms, which need a central node to collect all the relevant information (i.e., channel state information (CSI) and power parameters) and then solve the problem. This may not be appealing from a practical implementation perspective, especially when the amount of CSI needs to be shared is large (e.g., a large number of antennas or highly dispersive channels) and/or a central node is difficult to build. In [11], a distributed energy-efficient power optimization for SISO interference channels was studied. Although the main computational efforts can be done in a distributed manner at each BS, some information

This work was supported by the Academy of Finland under projects "WiFiUS: Message and CSI Sharing for Cellular Interference Management with Backhaul Constraints" and "Wireless Connectivity for Internet of Everything". It has been co-funded by the Irish Government and the European Union under Ireland's EU Structural and Investment Funds Programmes 2014-2020 through the SFI Research Centres Programme under Grant 13/RC/2077. This paper was presented in part at IEEE International Conference on Communications (ICC 2016), Kuala Lumpur, Malaysia, May 23–27, 2016 [1].

K.-G. Nguyen, Q.-D. Vu and M. Juntti are with Centre for Wireless Communications, University of Oulu, FI-90014, Finland (e-mail: {giang.nguyen, doanh.vu, markku.juntti}@oulu.fi).

L.-N. Tran is with School of Electrical and Electronic Engineering, University College Dublin, Ireland. He was with the Department of Electronic Engineering, Maynooth University, Ireland (e-mail: nam.tran@ucd.ie).

still needs to be collected and processed at the central node.

In this paper we consider multicell multiuser multiple-input single-output (MISO) downlink channels and propose two decentralized solutions for the problem of achieving fairness EE among BSs, each of which is limited by a total transmit power budget. The two proposed algorithms are a result of applying the SCA method to obtain convex approximations of the nonconvex max-min EE problem and applying the ADMM to solve the resulting convex programs. We note that in our recent work of [13], generic convex programs (GCPs) were derived to approximate the max-min EE problem. In a revised attempt, we introduce novel transformations to approximate the max-min EE problem into a series of second order cone programs (SOCPs). This leads to enormous reduction in terms of computational complexity as numerically demonstrated in Section VI of this paper. For the ease of description, the convex approximate problem obtained at iteration n of the SCA procedure is simply called the SCA$_n$ problem.

Motivated by many successful applications of the ADMM to numerous problems in wireless communications, e.g., in [15], [16], we aim to particularize the ADMM to solve the SCA$_n$ problem in a decentralized manner. However, the SCA$_n$ problem is not in a standard form that allows for a direct application of the ADMM. To make it possible we introduce the local and global versions of the so-called interference temperature, and decompose the SCA$_n$ problem into subproblems that can be solved independently at each BS. In the first proposed distributed method which strictly follows the ADMM, we allow the BSs to exchange the relevant information until the ADMM converges. In other words the ADMM is carried out until an optimal solution to the SCA$_n$ problem is found. Once this has been accomplished, all the BSs update the relevant parameters to create the SCA$_{n+1}$ problem for which another loop of the ADMM is carried out. In this way the first proposed algorithm can yield the same performance as the centralized method, but the amount of exchanged information is significant in some cases. To reduce the backhaul signaling overhead, we also consider a modified version of the first algorithm in which the ADMM is early terminated, i.e., after a small number of variable updates.

In the second proposed method we perform only *one* variables update in the ADMM. More specifically the BSs carry out only one iteration of the ADMM when solving the SCA$_n$ problem, and immediately consider the SCA$_{n+1}$ problem. The idea behind the second proposed solution is that, one iteration for the ADMM is sufficient when the SCA procedure nearly converges. This is true if a warm-start technique is used in the ADMM, i.e., the ADMM for the SCA$_{n+1}$ problem is initialized by the solution obtained from the SCA$_n$ problem. In view of this, the ADMM is not used to find an optimal solution of the SCA$_n$ problem, but to guide the local and global versions of the interference temperature towards the consensus value, while the SCA procedure is to update the interference temperature. The two proposed distributed solutions are provably convergent.

The rest of the paper is organized as follows. System model and the max-min EE problem are described in Section II. In Section III, a brief review of the centralized SCA based approach is presented, followed by the new novel transformation to approximate the max-min EE problem. Two proposed decentralized SCA-ADMM based algorithms are presented in Section IV. The convergence of the proposed algorithms is analyzed in Section V. Section VI provides the simulation results and concludes the paper.

Notation: The following notations are used throughout the paper. Vectors are denoted by bold lowercase letters and sets are denoted by calligraphic letters. $\mathbb{C}^{a\times b}$ represents the space of complex matrices of dimension given as superscripts. $(.)^T$ and $(.)^H$ represent the transpose and Hermitian transpose operator, respectively. $|.|$ and $\Re(.)$ represent the absolute value and real part of a complex number, respectively. $\|.\|_2$ represents the ℓ_2 norm. $\{\mathbf{x}_b\}_{b\in\mathcal{B}}$ refers to a composite vector that containing all $\mathbf{x}_b$ where b belongs to the set $\mathcal{B}$.

II. System Model and Problem Formulation

We consider a MISO downlink system consisting of B BSs, each equipped with N antennas. The set of BSs is denoted by $\mathcal{B}=\{1,\ldots,B\}$. To lighten the notation we assume each BS serves a different group of K single-antenna users and refer to the kth user served by BS b as b_k for $k=1,2,\ldots,K$. Note that the total number of users in the considered system is BK. Assuming a flat fading channel, the received signal at user b_k is

$$\begin{aligned} y_{b_k} = \mathbf{h}_{b,b_k}\mathbf{w}_{b_k}s_{b_k} + \textstyle\sum_{i=1,j\neq k}^{K}\mathbf{h}_{b,b_k}\mathbf{w}_{b_j}s_{b_j} \\ + \textstyle\sum_{i=1,i\neq b}^{B}\sum_{j=1}^{K}\mathbf{h}_{i,b_k}\mathbf{w}_{i_j}s_{i_j} + n_{b_k}, \end{aligned} \tag{1}$$

where $\mathbf{h}_{i,b_k}\in\mathbb{C}^{1\times N}$ is the channel vector from BS i to user b_k; $\mathbf{w}_{b_k}\in\mathbb{C}^{N\times 1}$ and s_{b_k} are the beamforming vector and the transmit data symbol from BS b to user b_k, respectively; $n_{b_k}\sim\mathcal{CN}(0,N_0)$ is the additive white Gaussian noise. We treat the inter-user interference as Gaussian noise and thus the SINR for user b_k is given by

$$\gamma_{b_k}(\mathbf{w}) = \frac{|\mathbf{h}_{b,b_k}\mathbf{w}_{b_k}|^2}{I_{b_k}(\mathbf{w})+WN_0}, \tag{2}$$

where W is the system bandwidth, $\mathbf{w}$ is the vector encompassing the beamformers of all BK users, and $I_{b_k}(\mathbf{w})$ is defined as

$$I_{b_k}(\mathbf{w}) \triangleq \textstyle\sum_{j=1,j\neq k}^{K}|\mathbf{h}_{b,b_k}\mathbf{w}_{b_j}|^2+\sum_{i=1,i\neq b}^{B}\sum_{j=1}^{K}|\mathbf{h}_{i,b_k}\mathbf{w}_{i_j}|^2. \tag{3}$$

Assuming Gaussian input signaling, the data rate of user b_k is given by $r_{b_k}(\mathbf{w})\triangleq W\log_2\big(1+\gamma_{b_k}(\mathbf{w})\big)$. In this paper, we are interested in achieving the EE fairness among individual BSs. Particularly, the EE of BS b, which is defined as the ratio of total data rate transmitted from BS b and its total power consumption [11], is expressed as

$$f_b(\mathbf{w}) = \frac{\sum_{k=1}^{K} r_{b_k}(\mathbf{w})}{\frac{1}{\epsilon}\sum_{k=1}^{K}\|\mathbf{w}_{b_k}\|_2^2 + NP_{\text{dp}}+P_{\text{sp}}} \tag{4}$$

where $\epsilon\in(0,1)$ is the power amplifier efficiency which is defined as the ratio between the total radio frequency (RF) output power and the direct current (DC) input power of the power amplifier (PA) [17]. Note that the value of ϵ depends on the design techniques and operating conditions of the PA

[17], [18]. In addition, P_{dp} is the dynamic power consumption corresponding to the power radiation of all circuit blocks in each active radio frequency chain; and P_{sp} is the static power spent by the cooling system, power supply, etc [14], [17], [18]. We denote by $P_0 \triangleq NP_{\text{dp}} + P_{\text{sp}}$ the total circuit power. The problem of max-min EE fairness among all the BSs is stated as

$$\max_{\mathbf{w}} \quad \min_{1\le b\le B} \; f_b(\mathbf{w}) \tag{5a}$$

$$\text{subject to} \quad \textstyle\sum_{k=1}^{K} \| \mathbf{w}_{b_k} \|_2^2 \le P_b, \quad \forall b \in \mathcal{B} \tag{5b}$$

where P_b is the transmit power budget of BS b.

To the best of our knowledge, the computational complexity of (5) has not been studied previously and seems to be challenging to characterize. Here we provide two remarks giving intuitive insights on its computational complexity corresponding two special scenarios. In particular, we discuss the complexity of (5) when the total circuit power P_0 is so large that maximizing the minimum EE amounts to maximizing the minimum of the sum rate of all BSs. That is to say, the denominator in (4) is mainly determined by P_0 which is a constant, and thus we only need to concentrate on maximizing the nominator. As a consequence, for the special case of $K = 1$, or a single user per BS (or commonly known as interference channel), this leads to the problem of SINR balancing which is known to be *polynomial time solvable* [19]. For the case when $K \ge 2$, the resulting problem is equivalent to the max-min utility problem with multiple tones which was proved to be *NP-hard* in [20, Theorem 2].

We remark that the existing solutions aim at finding a stationary solution to (5), which is a classical goal for general nonconvex problems [10], [12], [13]. Since (5) is basically a max-min fractional program, the so-called generalized Dinkelbach algorithm can be customized to solve (5) as done [10], [12]. Although it was shown in [21] that the induced parameter in Dinkelbach-type algorithms can converge linearly (or superlinearly under some stronger assumptions), these results are hardly applicable to (5). The main barrier is that the parametric programs obtained from applying the Dinkelbach's method to (5) are still nonconvex, and thus typically require a large number of iterations to find an optimal solution. To deal with this problem, such approaches as those in [10], [12] apply iterative local optimization methods, e.g. SCA or MMSE, to solve the parametric problems but their convergence rate may be relatively slow (cf. [13, Section IV]). As a result, these known algorithms are in fact a two-layer iterative procedure that needs a very high number of iterations to converge. Moreover, since these approaches can only achieve local optimality for each parametric problem, the convergence of the Dinkelbach's method may not be always guaranteed [8].

As an effort to reduce the computational complexity, we proposed in [13] a one-layer iterative method where (5) is reformulated into a more tractable representation which can be handled efficiently by the SCA framework. Nevertheless, the studies mentioned earlier arrived at centralized solutions, requiring a central node in their implementation to collect the required information and carry out all the computations, and thus are not practically appealing. Towards a distributed solution, we propose decentralized algorithms to solve the max-min EE problem in this paper. Particularly, our proposed approaches are inspired by the low-complexity SCA-based method in [13] whose details are given in the next section.

III. Centralized Beamforming Design for Max-Min EE Fairness

A. Equivalent Formulation

We start by introducing an equivalent reformulation of (5) which exposes the hidden convexity of the max-min EE problem. Specifically, (5) can be rewritten as

$$\max_{\mathbf{w},\eta,\mathbf{z},\mathbf{t}} \quad \eta \tag{6a}$$

$$\text{subject to} \quad z_b^2/t_b \ge \eta, \; \forall b \in \mathcal{B} \tag{6b}$$

$$\textstyle\sum_{k=1}^{K} \log(1 + \dfrac{|\mathbf{h}_{b,b_k}\mathbf{w}_{b_k}|^2}{I_{b_k}(\mathbf{w}) + WN_0}) \ge z_b^2, \; \forall b \in \mathcal{B} \tag{6c}$$

$$\textstyle\sum_{k=1}^{K} \| \mathbf{w}_{b_k} \|_2^2 / \epsilon + P_0 \le t_b, \; \forall b \in \mathcal{B} \tag{6d}$$

$$\textstyle\sum_{k=1}^{K} \| \mathbf{w}_{b_k} \|_2^2 \le P_b, \; \forall b \in \mathcal{B}, \tag{6e}$$

where η, $\mathbf{z} \triangleq [z_1, \ldots, z_B]^T \in \mathbb{R}^{B\times 1}$, $\mathbf{t} \triangleq [t_1, \ldots, t_B]^T \in \mathbb{R}^{B\times 1}$ are newly introduced slack variables. To deal with the nonconvexity of (6c) we further introduce new auxiliary variables $\mathbf{g} \triangleq [g_{1_1}, \ldots, g_{1_K}, g_{2_1}, \ldots g_{B_K}]^T \in \mathbb{R}^{BK\times 1}$ and $\mathbf{q} \triangleq [q_{1_1}, \ldots, q_{1_K}, q_{2_1}, \ldots q_{B_K}]^T \in \mathbb{R}^{BK\times 1}$, and rewrite (6) as

$$\max_{\mathbf{w},\eta,\mathbf{z},\mathbf{t},\mathbf{q},\mathbf{g}} \quad \eta \tag{7a}$$

$$\text{subject to} \quad z_b^2/t_b \ge \eta, \; \forall b \in \mathcal{B} \tag{7b}$$

$$\textstyle\sum_{k=1}^{K} \log(1 + g_{b_k}) \ge z_b^2, \; \forall b \in \mathcal{B}, \tag{7c}$$

$$|\mathbf{h}_{b,b_k}\mathbf{w}_{b_k}|^2/q_{b_k} \ge g_{b_k}, \; \forall b \in \mathcal{B}, k = 1, \ldots K \tag{7d}$$

$$I_{b_k}(\mathbf{w}) + WN_0 \le q_{b_k}, \; \forall b \in \mathcal{B}, k = 1, \ldots K \tag{7e}$$

$$\textstyle\sum_{k=1}^{K} \| \mathbf{w}_{b_k} \|_2^2 / \epsilon + P_0 \le t_b, \; \forall b \in \mathcal{B} \tag{7f}$$

$$\textstyle\sum_{k=1}^{K} \| \mathbf{w}_{b_k} \|_2^2 \le P_b, \; \forall b \in \mathcal{B}. \tag{7g}$$

It is easy to see that the constraints from (7b) to (7f) must hold with equality at the optimality. Thus, the equivalence between (5) and (7) is guaranteed. We further note that all the constraints listed in (7) are convex, excluding (7b) and (7d). This implies that we need to find ways to deal with the nonconvexity in (7b) and (7d).

B. New Centralized Approach

In this subsection we first briefly review the low-complexity beamforming design for the max-min EE problem introduced in our earlier work [13], then propose a new centralized solution that will be numerically shown to achieve a better performance when the system model is of small or moderate number of cells and users. In particular the solution presented in [13] was based on the observation that (7b) and (7d) admit the same form, i.e., the left-hand side (LHS) is a quadratic-over-affine function (which is convex) and the right-hand

side (RHS) is an affine function. This naturally leads to a direct application of the inner approximation algorithm [22] to approximate the nonconvex part of (7b) and (7d). More explicitly, we iteratively replace (7b) and (7d) by

$$\phi_b^n(z_b, t_b) \geq \eta,\ b \in \mathcal{B} \tag{8}$$

and

$$\psi_{b_k}^n(\mathbf{w}_{b_k}, q_{b_k}) \geq g_{b_k},\ b \in \mathcal{B}, k = 1, \dots K \tag{9}$$

respectively, where $\phi_b^n(z_b, t_b) \triangleq \frac{2z_b^n}{t_b^n} z_b - \frac{(z_b^n)^2}{(t_b^n)^2} t_b$ and $\psi_{b_k}^n(\mathbf{w}_{b_k}, q_{b_k}) \triangleq \frac{2\Re(\mathbf{h}_{b,b_k}^n \mathbf{w}_{b_k})}{q_{b_k}^n} - \frac{|\mathbf{h}_{b,b_k}\mathbf{w}_{b_k}^n|^2 q_{b_k}}{(q_{b_k}^n)^2}$ are the first order approximations of z_b^2/t_b and $|\mathbf{h}_{b,b_k}\mathbf{w}_{b_k}|^2/q_{b_k}$ around the point (z_b^n, t_b^n) and $(\mathbf{w}_{b_k}^n, q_{b_k}^n)$, respectively. We denote $\mathbf{h}_{b,b_k}^n = \left(\mathbf{w}_{b_k}^n\right)^H \mathbf{h}_{b,b_k}^H \mathbf{h}_{b,b_k}$. The superscript n denotes the nth iteration of the iterative procedure. In summary, the approximate convex program at iteration $n+1$ of the iterative algorithm proposed in [13] is given as

$$\max_{\mathbf{w},\eta,\mathbf{z},\mathbf{t},\mathbf{g},\mathbf{q}} \quad \{\eta \mid (7c), (7e), (7f), (7g), (8), (9)\}. \tag{10}$$

We note that $(z_b^n, t_b^n, \mathbf{w}_{b_k}^n, q_{b_k}^n)$ are treated as constants in (10) since they are the solution to the problem at iteration n. After solving (10), we update the involved variables for the next iteration until a stopping criterion is satisfied [13].

Convergence Analysis: Since the approximants of nonconvex constraints (7b) and (7d) meet the properties listed in [22], the proposed iterative procedure solving (10) produces a sequence of objective values that is provably monotonically convergent. Note that the solution obtained at each iteration satisfies the optimality conditions (i.e. the KKT conditions) of (10). Moreover, it is easy to check that the KKT conditions of (10) are indeed identical to those of (7) at convergence (cf. [22, Theorem 1]). In other words, a limit point of the proposed algorithm achieves the necessary optimality conditions of the original nonconvex problem (7).

We now turn our attention to the convex constraint in (7c) which is central to our discussions for the rest of this subsection. The idea in [13] was to preserve the convexity of (7c) and then approximate it (to a desired accuracy) by a system of small conic constraints. In this way the GCP in (10) can be cast as a second order cone program which can avail of state-of-the-art conic solvers such as MOSEK [23] or GUROBI [24]. In doing so, a number of slack variables are introduced to (10), which can result in an increase of per-iteration complexity. As an extension of our earlier work, we have numerically observed that in some settings, it is more computationally efficient to approximate (7c) by a constraint that can be SOC representable with a smaller number of additional slack variables. Toward this end we can see that (7c) is equivalent to the following two constraints

$$g_{b_k} \log(1 + g_{b_k}) \geq g_{b_k}\beta_{b_k},\ \forall b \in \mathcal{B}, k = 1, \dots K \tag{11a}$$

$$\textstyle\sum_{k=1}^K \beta_{b_k} \geq z_b^2,\ \forall b \in \mathcal{B} \tag{11b}$$

for $g_{b_k} > 0$, and $\boldsymbol{\beta} \triangleq [\beta_{1_1}, \dots, \beta_{1_K}, \beta_{2_1}, \dots \beta_{B_K}]^T \in \mathbb{R}^{BK\times 1}$ stands for the newly introduced slack variables. We note that the LHS of (11a) is convex with g_{b_k} which can be easily proved by checking the second order derivative of $g_{b_k}\log(1+g_{b_k})$. As a result the following inequality holds

$$\begin{aligned} g_{b_k}\log(1+g_{b_k}) \geq\ & g_{b_k}^n \log(1+g_{b_k}^n) + \\ & (g_{b_k} - g_{b_k}^n)(g_{b_k}^n(1+g_{b_k}^n)^{-1} + \log(1+g_{b_k}^n)) \\ =\ & g_{b_k} v_{b_k}^n - u_{b_k}^n,\ \forall b \in \mathcal{B}, k = 1, \dots K \end{aligned} \tag{12}$$

where $u_{b_k}^n \triangleq \frac{(g_{b_k}^n)^2}{g_{b_k}^n+1}$ and $v_{b_k}^n \triangleq \frac{g_{b_k}^n}{g_{b_k}^n+1} + \log(1+g_{b_k}^n)$. In light of SCA principle we now can replace (11a) with

$$g_{b_k} v_{b_k}^n - u_{b_k}^n \geq g_{b_k}\beta_{b_k},\ \forall b \in \mathcal{B}, k = 1, \dots K \tag{13}$$

which is equivalently represented by the following SOC constraint

$$\begin{aligned} g_{b_k} - \beta_{b_k} + v_{b_k}^n \geq \|[g_{b_k} + \beta_{b_k} - v_{b_k}^n \quad 2\sqrt{u_{b_k}^n}]\|_2, \\ \forall b \in \mathcal{B},\ k = 1, \dots K \end{aligned} \tag{14}$$

From the discussion from (11) to (14), we propose a new iterative algorithm, in which the convex problem at iteration $n+1$ reads as

$$\max_{\mathbf{w},\eta,\mathbf{z},\mathbf{t},\mathbf{g},\mathbf{q},\boldsymbol{\beta}} \quad \{\eta \mid (7e), (7f), (7g), (8), (9), (11b), (14)\} \tag{15}$$

The complexity of the proposed algorithm can be analyzed as follows. Recall that K, B, and N denote the number of users served by each BS, the number of BSs, and the number of antennas equipped for each BS, respectively. As can be seen, BK additional slack variables are introduced to approximate (7c) into the SOC constraints. According to the primal-dual path-following interior point method in [25, Section 6.6], the per-iteration complexity for solving (15) is $\mathcal{O}(N^3B^3K^3 + B^3K^3)$. Recall that the per-iteration complexity of the algorithm presented in [13] is $\mathcal{O}(N^3B^3K^3 + (m+7)^3B^3K^3)$, where m is the parameter that depends on the desired accuracy to approximate the exponential cone. In addition, the per-iteration complexity for solving the GCP (10) using (7c) is $\mathcal{O}(N^4B^4K^4 + B^4K^4)$. Obviously, the new algorithm based on (15) can reduce the per-iteration complexity as numerically shown in Figs. 2 and 3 in Section VI.

IV. Decentralized Approaches for the max-min EE

Our goal in this section is to propose decentralized solutions to solve (7). Those are of practical interest when a central processing station is not available, or when sending the CSI of all BSs to a central node is overwhelming. To the best of our knowledge, decentralized solutions for the max-min EE problem have not been reported previously in the related literature. The idea is to propose distributed approaches for solving the convex program obtained in each step of the iterative procedure. Thus, we may base our decentralized proposed algorithms on the SCA$_{n+1}$ problem given in (10) or in (15), depending on the problem size. Without loss of generality we represent a decentralized algorithm to solve (10) optimally in the sequel, and note that a decentralized method to solve (15) can be obtained similarly. Before proceeding further, we note that in this paper, CSI of all users is locally known at each BS, which is a standard assumption in numerous studies in the context of designing distributed algorithms for wireless

communications applications [15], [26]–[28]. More explicitly, each BS $b \in B$ perfectly knows the CSI of the channels from itself to all users in the network, i.e., $\{\mathbf{h}_{b,i_j}\}_{\forall i_j}$.

A. Pure ADMM based Decentralized Method for Max-Min EE

A simple and popular approach in the context of distributed optimization is the dual decomposition method. However, this approach cannot be applied to solve the SCA_{n+1} problem in (10) due to the lack of strict convexity of the objective (see [29] for further details). A similar problem was also mentioned in [15], but in a different context. A remedy for this issue is to combine the dual decomposition and the augmented Lagrangian method which results in the so-called ADMM [15], [29]. In addition to the lack of strict convexity, the formulation in (10) is not amendable to a direct application of ADMM since (7e) is not decomposable due to the intercell interference (ICI) contained in $I_{b_k}(\mathbf{w})$. To overcome these issues we rewrite (10) as

$$\min_{\substack{\mathbf{w},\eta,\{\eta_b\},\mathbf{z},\mathbf{t},\mathbf{g},\mathbf{q},\\ \{\tau_{b,i_j}\},\{\tau'_{b_k,i}\},\{\mu_{b,i_j}\}}} \quad -\sum_{b=1}^{B}\eta_b \tag{16a}$$

$$\text{subject to} \quad \tau_{b,i_j} \geq \textstyle\sum_{k=1}^{K}|\mathbf{h}_{b,i_j}\mathbf{w}_{b_k}|^2,\ \forall i \neq b \tag{16b}$$

$$q_{b_k} \geq \textstyle\sum_{j=1,j\neq k}^{K}|\mathbf{h}_{b,b_k}\mathbf{w}_{b_j}|^2 + \sum_{i=1,i\neq b}^{B}\tau'_{b_k,i} + WN_0, \tag{16c}$$

$$\eta_b = \eta,\ b \in \mathcal{B} \tag{16d}$$

$$\tau_{b,i_j} = \mu_{b,i_j}\ \forall i \neq b, j = 1,\dots K \tag{16e}$$

$$\tau'_{b_k,i} = \mu_{i,b_k}\ \forall i \neq b, k = 1,\dots K \tag{16f}$$

$$(7c), (7f), (7g), (8), (9), \tag{16g}$$

where η_b, τ_{b,i_j}, $\tau'_{b_k,i}$ are newly introduced slack variables, representing the energy efficiency of BS b, the ICI produced by BS b to user i_j, and the ICI that user b_k causes to neighboring BSs i ($i \neq b$), respectively. We remark that η_b, τ_{b,i_j}, $\tau'_{b_k,i}$ are locally stored at BS b. On the other hand, η, μ_{b,i_j} and μ_{i,b_k} are respectively the global copies of η_b, τ_{b,i_j} and $\tau'_{b_k,i}$. Intuitively, η_b is introduced to decouple the objective in (10), and (16d) is to force $\{\eta_b\}_{b\in\mathcal{B}}$ to be identical. Similarly, we introduce slack variables representing ICI and impose constraints (16e) and (16f) to decouple (7e). In this regard, τ_{b,i_j} and $\tau'_{i_j,b}$ both represent the value of the ICI produced by BS b to user i_j (i.e., $\sum_{k=1}^{K}|\mathbf{h}_{b,i_j}\mathbf{w}_{b_k}|^2$), where τ_{b,i_j} is the local variable stored by BS b and $\tau'_{i_j,b}$ is that of BS i. Thus the slack variable μ_{b,i_j} is to ensure τ_{b,i_j} and $\tau'_{i_j,b}$ are equal, i.e., $\tau_{b,i_j} = \mu_{b,i_j} = \tau'_{i_j,b}$, which is due to (16e) and (16f). It is easy to see that (16) is equivalent to (10). For the ease of description we will refer to the (local or global) variables representing interference as (local or global) interference temperature since their value varies with iterations. Next we include all the constraints that can be handled locally at BS b in the set $\mathcal{S}_b^{n+1}$ which is defined as

$$\mathcal{S}_b^{n+1} \triangleq \{\mathbf{s}_b \mid (7c), (7f), (7g), (8), (9), (16b), (16c)\} \tag{17}$$

where $\mathbf{s}_b \triangleq [\eta_b, z_b, t_b, g_{b_1}, \dots, g_{b_K}, \mathbf{w}_{b_1}, \dots \mathbf{w}_{b_K}, q_{b_1}, \dots q_{b_K}, \boldsymbol{\theta}_b^T]$ simply stacks all the local variables at BS b. Note that we have defined $\boldsymbol{\theta}_b \triangleq [\tau_{b,1_1}, \dots, \tau_{b,(b-1)_K}, \tau_{b,(b+1)_1}, \dots, \tau_{b,B_K}, \tau'_{b_1,1}, \dots, \tau'_{b_K,b-1}, \tau'_{b_1,b+1}, \dots, \tau'_{b_K,B}]^T$ which represents the local interference temperatures of BS b. Similarly we denote $\boldsymbol{\nu}_b \triangleq [\mu_{b,1_1}, \dots, \mu_{b,(b-1)_K}, \mu_{b,(b+1)_1}, \dots, \mu_{b,B_K}, \mu_{1,b_1}, \dots, \mu_{b-1,b_K}, \mu_{(b+1),b_1}, \dots, \mu_{B,b_K}]^T$, by sorting the corresponding global versions of $\boldsymbol{\theta}_b$. From the definitions above, (16) can be equivalently rewritten in a more compact form as

$$\min_{\mathbf{s},\boldsymbol{\vartheta}} \quad -\textstyle\sum_{b=1}^{B}\eta_b \tag{18a}$$

$$\text{subject to} \quad \mathbf{s}_b \in \mathcal{S}_b^{n+1},\ \forall b \in \mathcal{B} \tag{18b}$$

$$\eta_b = \eta,\ \forall b \in \mathcal{B} \tag{18c}$$

$$\boldsymbol{\theta}_b = \boldsymbol{\nu}_b,\ \forall b \in \mathcal{B} \tag{18d}$$

where $\mathbf{s} \triangleq [\mathbf{s}_1, \dots, \mathbf{s}_B]$ and $\boldsymbol{\vartheta} \triangleq [\eta, \boldsymbol{\nu}_1^T, \dots, \boldsymbol{\nu}_B^T]$. We are now in a position to describe the ADMM to solve (18) in a decentralized manner.

First, the augmented Lagrangian function of (18) is given by

$$\begin{aligned}\mathcal{L}^{n+1}(\boldsymbol{\vartheta}, \mathbf{s}, \boldsymbol{\zeta}, \boldsymbol{\xi}) =& \textstyle\sum_{b=1}^{B}[-\eta_b + \xi_b(\eta_b - \eta) + \boldsymbol{\zeta}_b^T(\boldsymbol{\theta}_b - \boldsymbol{\nu}_b) \\ &+ \frac{c}{2}((\eta_b - \eta)^2 + \|\boldsymbol{\theta}_b - \boldsymbol{\nu}_b\|_2^2)],\end{aligned} \tag{19}$$

where $\boldsymbol{\xi} \triangleq [\xi_1, \dots, \xi_B]$, $\boldsymbol{\zeta} \triangleq [\boldsymbol{\zeta}_1^T, \dots, \boldsymbol{\zeta}_B^T]$; ξ_b and $\boldsymbol{\zeta}_b$ are the Lagrange multipliers associated with (18c) and (18d), respectively. In (19), $c > 0$ is the penalty parameter and the quadratic penalty term $\frac{c}{2}((\eta_b - \eta)^2 + \|\boldsymbol{\theta}_b - \boldsymbol{\nu}_b\|_2^2)$ is to provide strict convexity w.r.t. $\mathbf{s}$ and $\boldsymbol{\vartheta}$ for (19), and thus problem (18) is always solvable [29]. The general idea of the ADMM is to use the Gauss-Seidel method to update the global variables (i.e., $\boldsymbol{\vartheta}$), the local variables (i.e., $\mathbf{s}$), and the Lagrange multipliers (i.e., $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$). Intuitively, at each iteration of the ADMM, all BSs agree a common knowledge of associated interference temperatures and the consensus EE by the update of global variables. Then, each BS independently solves its own subproblem and updates its Lagrange multipliers such that involved variables are driven into equality. Details of the proposed algorithm are presented as follows.

Update of Global Variables

An iteration of the first proposed algorithm starts with the update of global variables, which can be done by fixing the local variables when solving the optimization problem $\min_{\boldsymbol{\vartheta}}\{\mathcal{L}^{n+1}(\boldsymbol{\vartheta}, \mathbf{s}^l, \boldsymbol{\zeta}^l, \boldsymbol{\xi}^l)\}$. Note that the superscript associated with each variable is the iteration counter of the ADMM part. Since $\mathcal{L}^{n+1}(\boldsymbol{\vartheta}, \mathbf{s}^l, \boldsymbol{\zeta}^l, \boldsymbol{\xi}^l)$ is separable in η and $\{\boldsymbol{\nu}_b\}$, the $\boldsymbol{\vartheta}$-minimization step leads to the following independent subproblems

$$\eta^{l+1} = \arg\min_{\eta} \textstyle\sum_{b=1}^{B}(-\xi_b^l\eta + \frac{c}{2}(\eta_b^l - \eta)^2) \tag{20}$$

and

$$\boldsymbol{\nu}_b^{l+1} = \arg\min_{\boldsymbol{\nu}_b} -(\boldsymbol{\zeta}_b^l)^T\boldsymbol{\nu}_b + \frac{c}{2}\|\boldsymbol{\theta}_b^l - \boldsymbol{\nu}_b\|_2^2, \quad \forall b \in \mathcal{B}. \tag{21}$$

Note that in above equation η_b^l and $\boldsymbol{\theta}_b^l$ are the values obtained after solving (25) at iteration l and are exchanged by all BSs.

We also remark that (20) and (21) are unconstrained quadratic programs for which the closed form solutions are given by

$$\eta^{l+1} = \textstyle\sum_{b=1}^{B}(\eta_b^l + \xi_b^l/c)/B \tag{22}$$

and

$$\begin{aligned}\mu_{b,i_j}^{l+1} =&(\tau_{b,i_j}^l + \tau_{i_j,b}'^l)/2 + (\zeta_{b,i_j}^l + \zeta_{i_j,b}^l)/2c\\ \mu_{i,b_k}^{l+1} =&(\tau_{i,b_k}^l + \tau_{b_k,i}'^l)/2 + (\zeta_{i,b_k}^l + \zeta_{b_k,i}^l)/2c\end{aligned} \tag{23}$$

where $\{\zeta_{b,i_j}, \zeta_{b_k,i}\}$ and $\{\zeta_{i,b_k}, \zeta_{i_j,b}\}$ are the dual variables associated to the primal variables $\{\tau_{b,i_j}, \tau'_{b_k,i}\}$ of BS b and $\{\tau_{i,b_k}, \tau'_{i_j,b}\}$ of BS $i \neq b$, respectively. In addition, (22) and (23) are independently carried out at each BS so that η^{l+1} and $\boldsymbol{\nu}_b^{l+1}$ are available at BS b to update its local variables in the next step of the procedure. Particularly, to compute η^{l+1} in (22), all BSs can run an average consensus algorithm [3], [29], [30]. We note that μ_{b,i_j}^{l+1} and μ_{i,b_k}^{l+1} are computed after gathering $(\frac{\tau_{i_j,b}'^l}{2} + \frac{\zeta_{i_j,b}^l}{2c})$ and $(\frac{\tau_{i,b_k}^l}{2} + \frac{\zeta_{i,b_k}^l}{2c})$ from BS $i \neq b$, respectively. Since the required information is real-valued, the amount of exchanged signaling overhead is significantly lower, compared to directly sharing the complex-value CSI.

Update of Local Variables

The set of local variables $\mathbf{s}$ is updated by solving the convex problem

$$\mathbf{s}^{l+1} = \underset{\mathbf{s}_b \in \mathcal{S}_b^{n+1}, \forall b \in \mathcal{B}}{\operatorname{argmin}} \quad \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}, \boldsymbol{\xi}^l, \boldsymbol{\zeta}^l). \tag{24}$$

The main point of the ADMM is that the augmented Lagrangian function in (19) is decomposable in $\mathbf{s}_b$, and thus the optimization problem (24) can be solved in parallel at each BS. Specifically, BS b can locally solve the following convex subproblem

$$\begin{aligned}\mathbf{s}_b^{l+1} = \underset{\mathbf{s}_b \in \mathcal{S}_b^{n+1}}{\arg\min} & -\eta_b + \xi_b^l(\eta_b - \eta^{l+1}) + (\boldsymbol{\zeta}_b^l)^T(\boldsymbol{\theta}_b - \boldsymbol{\nu}_b^{l+1})\\ &+ \frac{c}{2}\left((\eta_b - \eta^{l+1})^2 + \|\boldsymbol{\theta}_b - \boldsymbol{\nu}_b^{l+1}\|_2^2\right)\end{aligned} \tag{25}$$

using only its local CSI, i.e., $\{\mathbf{h}_{b,i_j}\}_{i_j}$, and η^{l+1} and $\boldsymbol{\nu}_b^{l+1}$ from the update of global variables. We recall that all the constraints defining the set $\mathcal{S}_b^{n+1}$ are SOC representable, and thus (25) can be efficiently solved by SOCP solvers.

Update of Lagrange Multipliers

The last step of the ADMM is to update the Lagrange multipliers as

$$\xi_b^{l+1} = \xi_b^l + c(\eta_b^{l+1} - \eta^{l+1}),\ \forall b \in \mathcal{B} \tag{26}$$

$$\boldsymbol{\zeta}_b^{l+1} = \boldsymbol{\zeta}_b^l + c(\boldsymbol{\theta}_b^{l+1} - \boldsymbol{\nu}_b^{l+1}),\ \forall b \in \mathcal{B}. \tag{27}$$

Because the current iterates of the local and global variables are already available to all BSs at this step, the update of the Lagrange multipliers does not require extra information exchange, and thus incurs no signaling overhead.

After the ADMM procedure converges, i.e., the primal residual $\varepsilon_{\text{ADMM}}$ (see step 3 of Algorithm 1) is below a threshold, we update the involved SCA parameters, i.e., $(\mathbf{w}^{n+1}, \mathbf{q}^{n+1}, \mathbf{z}^{n+1}, \mathbf{t}^{n+1})$ as in the centralized algorithm and carry out another ADMM step to solve the SCA_{n+2}. This iterative process is terminated until the SCA outer loop converges. To summarize, Algorithm 1 outlines the proposed decentralized algorithm for the max-min EE beamforming design problem. We now discuss some practical issues and complexity analysis in relation to the implementation of Algorithm 1.

Algorithm 1 Proposed decentralized beamformer ADMM-based design for max-min EE in multicell multiuser MISO downlink

1: **Initialization:** Set $n := 0$, $l := 0$ and choose feasible initial values for $(\mathbf{w}^n, \mathbf{q}^n, \mathbf{z}^n, \mathbf{t}^n)$ and choose the initial values for $\{\eta_b^0\}$, $\{\boldsymbol{\theta}_b^0\}$, $\boldsymbol{\xi}^0$, $\boldsymbol{\zeta}^0$.
2: **repeat** {SCA outer loop}
3: **while** $\varepsilon_{\text{ADMM}} \triangleq \sqrt{\sum_{b=1}^{B} \|[\eta_b^l, (\boldsymbol{\theta}_b^l)^T]^T - [\eta^l, (\boldsymbol{\nu}_b^l)^T]^T\|_2^2} \geq 10^{-5}$ **do** {ADMM inner loop}
4: **for** $b \in \mathcal{B}$ **do**
5: BS b updates η^{l+1} through an average consensus algorithm [30].
6: BS b receives $(\frac{\tau_{i_j,b}'^l}{2} + \frac{\zeta_{i_j,b}^l}{2c})$ and $(\frac{\tau_{i,b_k}^l}{2} + \frac{\zeta_{i,b_k}^l}{2c})$ then updates μ_{b,i_j}^{l+1} and μ_{i,b_k}^{l+1} by (23).
7: BS b updates $\mathbf{s}_b^{l+1}$ by (25).
8: Update Lagrange multipliers ξ_b^{l+1} and $\boldsymbol{\zeta}_b^{l+1}$ by (26) and (27), respectively.
9: **end for**
10: $l := l + 1$.
11: **end while**
12: Obtain the optimal value $(\mathbf{w}^*, \mathbf{q}^*, \mathbf{z}^*, \mathbf{t}^*)$.
13: Update the SCA parameters $(\mathbf{w}^{n+1}, \mathbf{q}^{n+1}, \mathbf{z}^{n+1}, \mathbf{t}^{n+1}) = (\mathbf{w}^*, \mathbf{q}^*, \mathbf{z}^*, \mathbf{t}^*)$, and $(\{\eta_b^0\}, \{\boldsymbol{\theta}_b^0\}, \boldsymbol{\xi}^0, \boldsymbol{\zeta}^0) = (\{\eta_b^*\}, \{\boldsymbol{\theta}_b^*\}, \boldsymbol{\xi}^*, \boldsymbol{\zeta}^*)$.
14: $n := n + 1$; $l := 0$.
15: **until** the SCA converges

1) Amount of Exchanged Information: Algorithm 1 requires BS b to broadcast $\boldsymbol{\theta}_b$ and η_b to the other BSs. The amount of backhaul signaling mainly depends on steps 5 and 6 of Algorithm 1 as can be seen in Table I. To update global variables in steps 5 and 6, BS b needs to send out 2 and $2K$ real values to other $(B-1)$ BSs, respectively. In the current LTE systems, the information sharing process among BSs can be done via the X2 interface.

2) Per-BS Complexity Analysis: We remark that the per-iteration complexity of Algorithm 1 is dominated by the complexity of solving the subproblem (25) at each BS. As presented in the previous section the constraint $\sum_{k=1}^{K} \log(1 + g_{b_k}) \geq z_b^2$ in the set $\mathcal{S}_b^{n+1}$ can be handled by two ways, one based on (12) and the other based on the SOC approximation of the exponential cone in [13, Eq. (13)]. Thus, the cost of solving the subproblem (25) with the former and latter approximation is respectively $\mathcal{O}(N^3K^3)$ and $\mathcal{O}(N^3K^3 + (m+7)^3K^3)$ [25, Sect. 6.6], where m is the conic approximation parameter of accuracy. The choice of a proper approximation depends on the problem size which is numerically discussed in Figs. 2 and 3.

3) Improved Convergence Rate: We now present a variant of Algorithm 1 which may potentially improve its convergence

Table I
INFORMATION EXCHANGE IN ALGORITHM 1.

Procedure	Exchanged information
Step 5, update η at BS b	$\{\eta_b\}$ and $\{\xi_b\}$ (average consensus among BSs)
Step 6, update μ_{b,i_j} and μ_{i,b_k} at BS b	$(\frac{\tau'^l_{i_j,b}}{2}+\frac{\zeta^l_{i_j,b}}{2c})$ and $(\frac{\tau^l_{i,b_k}}{2}+\frac{\zeta^l_{i,b_k}}{2c})$ from BS $i\neq b$

rate in practice. Note that waiting for the ADMM-based loop to completely converge at each SCA iteration ensures the global convergence of Algorithm 1 but generally slows down its convergence rate. The idea is to allow the ADMM part to terminate early when solving the SCA_n for some first n. In fact, we numerically obverse that the increase in the cost function between two consecutive iterations is large in some first SCA iterations, and thus it usually requires a large number of variable updates for the ADMM to converge. However, the solutions returned by the ADMM part are still rough estimates of the solution of the max-min EE problem in the first SCA problems. Thus, letting the ADMM to completely converge in the first stage of the SCA loop is usually not beneficial. To improve the convergence rate we stop the ADMM loop after a fixed number of updates, say I_{ADMM}. That is, the ADMM inner loop is terminated if $l > \text{I}_{\text{ADMM}}$, and the value of I_{ADMM} can be varied as the SCA outer loop evolves. For example, I_{ADMM} can be set to be higher for some first SCA problems so that the ADMM part can produce a good estimate of the solution. On the other hand, when the SCA is nearly convergent, we can set I_{ADMM} to be smaller. By adapting I_{ADMM} we can enhance the convergence rate of Algorithm 1 significantly as shown in Section VI. The special case when $\text{I}_{\text{ADMM}} = 1$ for all SCA iterations deserves particular attention and is treated in the following subsection.

B. Interlaced SCA-ADMM based Decentralized Method for Max-Min EE

A special variant of Algorithm 1 is described in Algorithm 2, which is obtained by carrying out the next SCA iteration immediately after each ADMM iteration, rather than letting the ADMM completely converge. Thereby the SCA and the ADMM iterations are tightly interlaced and merged into a single layer iterative algorithm. It is best to view Algorithm 2 as a result of applying the ADMM to the original nonconvex problem in (7), but the local feasible sets are approximated by the SCA principle at each ADMM update. To be more specific, let us define the (nonconvex) feasible set of $\{\mathcal{S}_b\}$ for BS b as $\mathcal{S}_b \triangleq \{\mathbf{s}_b \mid (7\text{b}), (7\text{c}), (7\text{d}), (7\text{f}), (7\text{g}), (16\text{b}), (16\text{c})\}$. Note that $\mathcal{S}_b$ is the original feasible set at BS b where the nonconvex constraints are *not* approximated by the convex approximants using the SCA principle. Then we can equivalently rewrite (18) as

$$\min_{\mathbf{s},\boldsymbol{\vartheta}}\{-\textstyle\sum_{b=1}^{B}\eta_b \mid \mathbf{s}_b\in\mathcal{S}_b,\ \eta_b=\eta,\ \boldsymbol{\theta}_b=\boldsymbol{\nu}_b,\ \forall b\in\mathcal{B}\}. \quad (28)$$

The difference between (28) and (18) is that $\mathcal{S}_b$ is used in (28) instead of a convex approximation $\mathcal{S}_b^{n+1}$ in (18). The augmented Lagrangian for (28) is given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\vartheta},\mathbf{s},\boldsymbol{\zeta},\boldsymbol{\xi}) = \textstyle\sum_{b=1}^{B}[-\eta_b+\xi_b(\eta_b-\eta)+\boldsymbol{\zeta}_b^T(\boldsymbol{\theta}_b-\boldsymbol{\nu}_b)\\ +\frac{c}{2}((\eta_b-\eta)^2+\|\boldsymbol{\theta}_b-\boldsymbol{\nu}_b\|_2^2)].\end{aligned} \quad (29)$$

The ADMM in Algorithm 2 has the form

$$\boldsymbol{\vartheta}^{l+1} = \arg\min_{\boldsymbol{\vartheta}}\ \mathcal{L}(\boldsymbol{\vartheta},\mathbf{s}^l,\boldsymbol{\zeta}^l,\boldsymbol{\xi}^l), \quad (30a)$$

$$\mathbf{s}_b^{l+1} = \arg\min_{\mathbf{s}_b\in\mathcal{S}_b^{l+1}}\ \mathcal{L}(\boldsymbol{\vartheta}^{l+1},\mathbf{s},\boldsymbol{\zeta}^l,\boldsymbol{\xi}^l),\ \forall b\in\mathcal{B}, \quad (30b)$$

$$\begin{aligned}\xi_b^{l+1} &= \xi_b^l + c(\eta_b^{l+1}-\eta^{l+1})\\ \boldsymbol{\zeta}_b^{l+1} &= \boldsymbol{\zeta}_b^l + c(\boldsymbol{\theta}_b^{l+1}-\boldsymbol{\nu}_b^{l+1})\end{aligned},\forall b\in\mathcal{B}. \quad (30c)$$

Note that the update of global variables in (30a) is done exactly as in Algorithm 1, i.e., in (20) and (21). Particularly, the update of local variables at iteration $l+1$ is carried out *inexactly*. That is the $\mathbf{s}_b$-minimization in (30b) is over the approximate convex set $\mathcal{S}_b^{l+1}$ defined in (17) which arises from the SCA framework, not over the original nonconvex set $\mathcal{S}_b$. Intuitively, the ADMM part guides the global and local variables towards an equality and the SCA is used to find a good locally feasible solution. The convergence of Algorithm 2 is provable as shown in Section V.

We note that the analysis of the amount of exchanged information and the per-BS complexity at each iteration are the same as those in Algorithm 1, since the size of the problem and the amount of required exchanged information do not change.

Algorithm 2 Proposed interlaced SCA-ADMM based decentralized algorithm for Max-Min EE in multicell multiuser MISO downlink

1: **Initialization:** Set $l := 0$, choose feasible initial values for $(\mathbf{w}^l, \mathbf{q}^l, \mathbf{z}^l, \mathbf{t}^l)$ and choose the initial values for $\{\eta_b^0\}$, $\{\boldsymbol{\theta}_b^0\}$ $\boldsymbol{\xi}^0$, $\boldsymbol{\zeta}^0$.
2: **repeat**
3: **for** $b\in\mathcal{B}$ **do**
4: BS b updates η^{l+1} through an average consensus algorithm [30].
5: BS b receives $(\frac{\tau'^l_{i_j,b}}{2}+\frac{\zeta^l_{i_j,b}}{2c})$ and $(\frac{\tau^l_{i,b_k}}{2}+\frac{\zeta^l_{i,b_k}}{2c})$ then updates μ_{b,i_j}^{l+1} and μ_{i,b_k}^{l+1} by (23).
6: BS b updates $\mathbf{s}_b^{l+1}$ by (30b) by the approximate convex set $\mathcal{S}_b^{l+1}$.
7: Update Lagrange multipliers ξ_b^{l+1} and $\boldsymbol{\zeta}_b^{l+1}$ by (30c).
8: **end for**
9: Update the SCA parameters $(\mathbf{w}^{l+1}, \mathbf{q}^{l+1}, \mathbf{z}^{l+1}, \mathbf{t}^{l+1}) = (\mathbf{w}^*, \mathbf{q}^*, \mathbf{z}^*, \mathbf{t}^*)$.
10: $l := l+1$
11: **until** Convergence

It will be numerically shown in Section VI that Algorithm 2 may converge faster than Algorithm 1.

V. Convergence Analysis of Proposed Algorithms

In this section we provide the convergence analysis of the two algorithms proposed in the preceding section. Our simulation analysis is based on [31], in which the augmented Lagrangian is proved to be monotonically decreasing.

A. Convergence Proof of Algorithm 1

The convergence of Algorithm 1 is guaranteed by that of the ADMM and SCA procedures. First we discuss the convergence of the ADMM for solving the SCA_{n+1} problem. Recall that (16) is equal to (10) at the optimality. Thus, it is trivial to see that the set $\mathcal{S}_b^{n+1}$ defined in (17) is nonempty and the problem (18) is feasible. As a result, the subproblems (20), (21), and (25) are always solvable, i.e., the optimal solutions do exist and the optimal objectives are bounded below. We can also easily check that there exists a saddle point for the unaugmented Lagrangian, i.e., the function obtained by setting $c = 0$ in (19). Thus, the convergence of the ADMM part in Algorithm 1 is guaranteed due to the result in [29, Section 3.2]. We now provide another convergence result for the ADMM part in Algorithm 1, which can be used to prove the convergence of Algorithm 2. In particular, we have the following lemma.

Lemma 1. *The difference of the augmented Lagrangian function between two consecutive iterations in the ADMM inner loop of Algorithm 1 is*

$$\begin{aligned}&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1},\mathbf{s}^{l+1},\boldsymbol{\zeta}^{l+1},\boldsymbol{\xi}^{l+1})-\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})\\&\leq\sum_{b=1}^{B}\frac{1}{c}(d^2-\frac{c^2}{2})\left((\eta_b^{l+1}-\eta_b^{l})^2+\|\boldsymbol{\theta}_b^{l+1}-\boldsymbol{\theta}_b^{l}\|_2^2\right)\\&\quad-\sum_{b=1}^{B}\frac{c}{2}\left((\eta^{l+1}-\eta^{l})^2+\|\boldsymbol{\nu}_b^{l+1}-\boldsymbol{\nu}_b^{l}\|_2^2\right)\end{aligned}\tag{31}$$

where c is the penalty parameter and d is a positive constant which depends on the problem data.

The proof of Lemma 1 is deferred to Appendix A. Lemma 1 gives a useful insight into the convergence of the ADMM inner loop of Algorithm 1. That is, if c is chosen to be sufficiently large, i.e., $c > d\sqrt{2}$, then the Lagrangian function $\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})$ generated by the ADMM loop is monotonically decreasing. Note that this is a new convergence result for the ADMM applied to the considered problem, compared to the standard result in [29].

We now show that the SCA iteration is also provably convergent which follows the same line of argument as that in [22]. Towards this end let us denote the feasible set and the optimal solution, obtained by the ADMM procedure of the SCA_n problem is $\mathcal{S}^n$ and $\mathbf{s}^n$, respectively. The convergence analysis of the SCA outer loop is presented as follows. Due to the use of approximations in (8) and (9), $\mathbf{s}^n$ satisfies all the constraints in (18) at iteration $n+1$, i.e., $\mathbf{s}^n \in \mathcal{S}^{n+1}$. This immediately implies that the sequence of the objective values $\{\eta^n\}$ is non-increasing. Moreover, we can easily see that the power constraint in (5b) ensures that $\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})$ is bounded below for all l. Thus, the convergence of $\{\eta^n\}$ is established.

B. Convergence Proof of Algorithm 2

Following the new convergence result for the ADMM of Algorithm 1, we have the following lemma regarding the convergence of Algorithm 2.

Lemma 2. *The difference of the augmented Lagrangian function between two consecutive iterations in Algorithm 2 is*

$$\begin{aligned}&\mathcal{L}(\boldsymbol{\vartheta}^{l+1},\mathbf{s}^{l+1},\boldsymbol{\zeta}^{l+1},\boldsymbol{\xi}^{l+1})-\mathcal{L}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})\\&\leq\sum_{b=1}^{B}\frac{1}{c}(\tilde{d}^2-\frac{c^2}{2})\left((\eta_b^{l+1}-\eta_b^{l})^2+\|\boldsymbol{\theta}_b^{l+1}-\boldsymbol{\theta}_b^{l}\|_2^2\right)\\&\quad-\sum_{b=1}^{B}\frac{c}{2}\left((\eta^{l+1}-\eta^{l})^2+\|\boldsymbol{\nu}_b^{l+1}-\boldsymbol{\nu}_b^{l}\|_2^2\right)-\sum_{b=1}^{B}\delta_b^l\end{aligned}\tag{32}$$

where c is the penalty parameter, $\tilde{d}$ is a positive constant which depends on the problem data and

$$\begin{aligned}\delta_b^l=\;&\pi_b t_b^l\left(\frac{z_b^l}{t_b^l}-\frac{z_b^{l-1}}{t_b^{l-1}}\right)^2\\&+\sum_{k=1}^{K}\omega_{b_k}q_{b_k}^l\left(\frac{\Re(\mathbf{h}_{b,b_k}^l\mathbf{w}_{b,b_k}^l)}{q_{b_k}^l}-\frac{\Re(\mathbf{h}_{b,b_k}^{l-1}\mathbf{w}_{b,b_k}^{l-1})}{q_{b_k}^{l-1}}\right)^2,\end{aligned}$$

where $\pi_b \geq 0$ and $\omega_{b_k} \geq 0$ are dual variables associated to the approximated constraints when solving (30b)*.*

The proof of Lemma 2 is given in Appendix B and its corollary is discussed in the following remark.

Remark 3. We can see that the SCA step introduces the term $-\sum_{b=1}^{B}\delta_b^l \leq 0$ in the RHS of (32). Thus, if c is chosen to be large enough, i.e., $c > \tilde{d}\sqrt{2}$, then the RHS of (32) becomes negative, meaning that the augmented Lagrangian function $\mathcal{L}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})$ is monotonically decreasing. Recall that $\mathcal{L}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})$ is bounded below by the power constraint in (5b) for all l, and thus guaranteed to converge. Convergence analysis of Algorithm 2 for the case of small c is beyond the scope of this paper and is left for future work. Since it is hard to analytically derive the constant $\tilde{d}$, a practical way is to vary the value of c, i.e., iteratively increasing c until monotonic convergence of $\mathcal{L}(\boldsymbol{\vartheta}^{l},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})$ is established [29].

C. The Choice of Penalty Parameter c

The two lemmas presented above imply that the value of c has a significant impact on the convergence behavior of the proposed algorithms. On one hand, a large value of c should be considered to ensure the convergence of Algorithms 1 and 2. On the other hand, it was shown that better convergence may be achieved for small c [32]. This issue is numerically shown in Fig. 4 in Section VI.

We now present a way to further increase the convergence rate of Algorithms 1 and 2. Note that the values of η and $\{\eta_b\}$ are practically much smaller than those of $\{\boldsymbol{\theta}_b\}$ and $\{\boldsymbol{\nu}_b\}$. As a result the range of $(\eta_b-\eta)^2$ is much smaller than that

Table II
SIMULATION PARAMETERS [10], [11]

PARAMETERS	VALUE
Path loss and shadowing	$38\log_{10}(d_{b,i_j}\text{m}) + 34.5 + \mathcal{N}(0,8)$ dB
Inter-BS distance	$D = 1$ km
Static power consumption P_{sp}	33 dBm
Power amplifier efficiency ϵ	0.35
Number of BSs B	3
Number of Tx antennas N	4
Signal bandwidth W	10 kHz
Power spectral density of noise	-174 dBm/Hz

of $\|\boldsymbol{\theta}_b - \boldsymbol{\nu}_b\|_2^2$ during the iterative procedure. This imbalance between the two terms can cause a detrimental effect on the convergence rate of Algorithms 1 and 2, since the two residual terms tend to converge at different rates as they both approach zero [29, Sect. 3.4.1]. To overcome this issue, we make a simple modification to enhance the convergence speed of the proposed algorithms by maintaining the balance among the penalty terms in the Lagrangian. To this end we assign two different penalty parameters c_1 and c_2 to the two penalty terms in the augmented Lagrangian as

$$\mathcal{L}^{n+1}(\boldsymbol{\vartheta}, \mathbf{s}, \boldsymbol{\zeta}, \boldsymbol{\xi}) = \sum_{b=1}^{B}[-\eta_b + \xi_b(\eta_b - \eta) + \boldsymbol{\zeta}_b^T(\boldsymbol{\theta}_b - \boldsymbol{\nu}_b) + \frac{c_1}{2}(\eta_b - \eta)^2 + \frac{c_2}{2}\|\boldsymbol{\theta}_b - \boldsymbol{\nu}_b\|_2^2] \tag{33}$$

It is trivial to see that the convergence of Algorithms 1 and 2 is still guaranteed if c_1 and c_2 are sufficiently large. From the discussion above we will choose $c_1 > c_2$ to keep the two residual norms within a factor of one another as they both converge to zero [29, Sect. 3.4.1]. Numerical results on the convergence properties of Algorithms 1 and 2 are provided in Figs. 4 and 5 in the next section.

VI. NUMERICAL RESULTS

In this section we numerically evaluate the effectiveness of the proposed methods. The general simulation parameters are listed in Table II and specific ones are given in the caption of the corresponding figures. We consider the scenario where K users are randomly distributed in a cell with a radius of 500 m. For instance, the simulation scenario for a specific case where $B = 3$ and $K = 2$ is illustrated as in Fig. 1. The flat fading channel $\mathbf{h}_{b,i_j}$ is assumed to be Gaussian distributed, i.e., $\mathbf{h}_{b,i_j} \sim \mathcal{CN}(0, \rho_{b,i_j}\mathbf{I}_N)$ where ρ_{b,i_j} represents the path loss and shadowing between BS b and user i_j as given in Table II. The initial values of all proposed algorithms are generated as follows. First, a set of beamformers that satisfy (7g) is created at each BS. Then the initial values of other variables $(z_b^0, t_b^0, q_{b_k}^0)$ are obtained by setting all the constrains in (17) to be equality. The initial local values $\{\eta_b^0\}$ and $\{\boldsymbol{\theta}_b^0\}$ are calculated by the initial local variables set $\{\mathcal{S}_b^0\}$, and the Lagrange multipliers $\boldsymbol{\xi}^0$ and $\boldsymbol{\zeta}^0$ are simply set to zero. The iterative procedure of the proposed decentralized algorithms is terminated if the increase in η in two subsequent SCA iterations is less than 10^{-5}. We model the algorithms in MATLAB environment and all the convex problems considered in this paper are solved using MOSEK, a state-of-the-art solver for SOCPs.

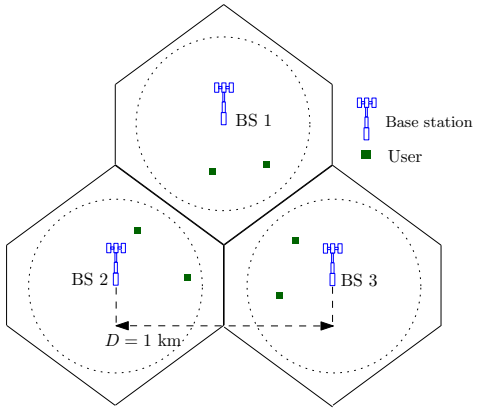


Fig. 1. Illustration of simulation scenario where $B = 3$ and $K = 2$. Users are randomly distributed within the dash circle of coverage region of each BS.

A. Convergence Results of New Approximation in (15)

We first provide a numerical comparison of the convergence results among the new formulation in (15), the one in our previous work of [13] and the GCP (10) with (7c) in Figs. 2 and 3. Specifically, we consider two scenarios with $B = 3$ and $B = 4$ BSs. Each point of the curves in Figs. 2 and 3(b) is obtained by averaging out 500 channel realizations. As expected, the use of the approximation methods reduces the per-iteration complexity compared to that of using the original non-linear constraint (7c). Furthermore, the new formulation in (15) can potentially reduce the per-iteration complexity compared to the one in [13]. This is confirmed by the numerical results shown in Fig. 2 where we report the average runtime for each iteration of the three schemes in comparison. Also, it is natural to imply that the new approach outperforms the one in [13], in terms of overall runtime. However this is not always the case as the overall runtime of the two schemes critically depends on the number of iterations required for convergence, which is studied in the next numerical experiment.

In Fig. 3(a), we compare the convergence rate of the three methods for a random channel realization with different numbers of users per BS K. We observe that the method in [13, eq. (13)] converges faster in terms of the required number of iterations, compared to the one proposed in (15), while resulting in the same convergence rate compared to solving the GCP (10). This is understandable because (12) is a lower

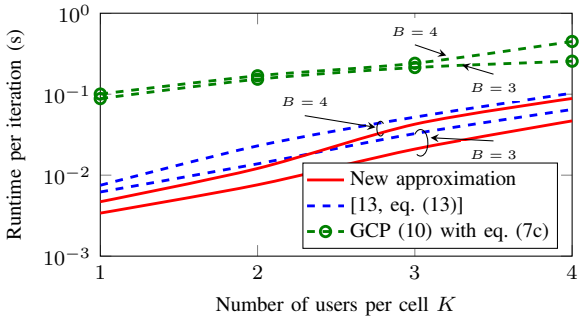
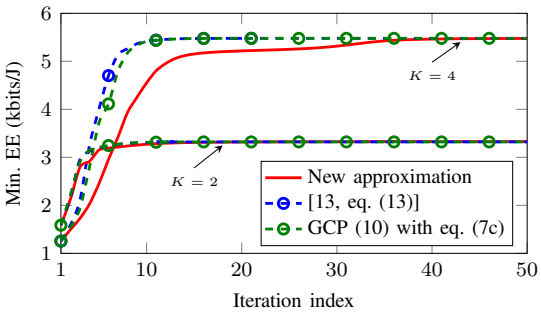
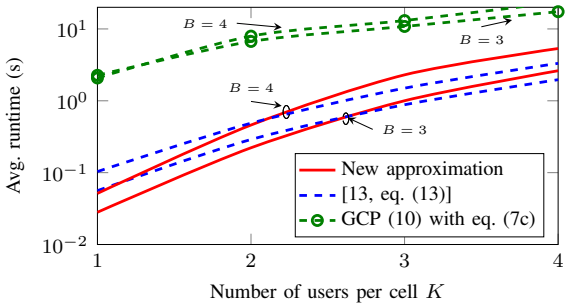


Fig. 2. Per-iteration runtime with $P_b = 35$ dBm and $P_{\text{dyn}} = 40$ dBm.



(a) Convergence rate with $B = 3$.



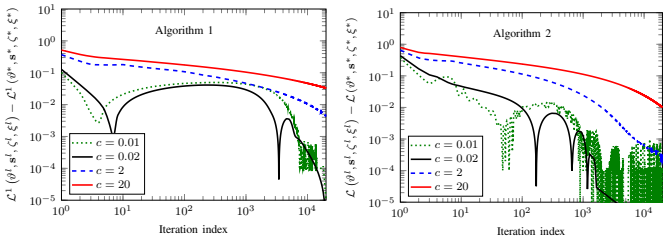
(b) Average runtime.

Fig. 3. Comparison of convergence rate and average runtime between new approximation algorithm, the one in [13] and GCP (10) with eq. (7c), with $P_b = 35$ dBm and $P_{\text{dyn}} = 40$ dBm.

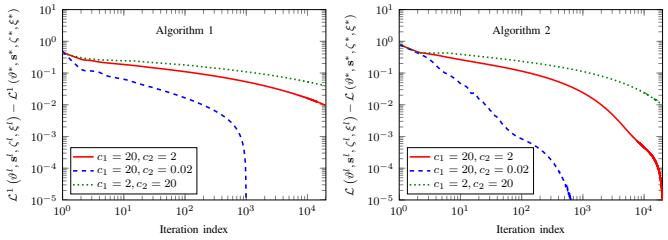
bound of $g_{b_k} \log(1 + g_{b_k})$, and the tightness of the bound evolves with iterations. On the contrary, the conic constraints in [13, eq. (13)] are an approximation of $\log(1 + g_{b_k})$ to a desired accuracy level, which is fixed during the iterative procedure. Consequently, the approximation in (12) usually takes more iterations to converge, especially when the problem size grows. This results in the average runtime of both approximation methods shown in Fig. 3(b). We can see that the formulation in (15) is more computationally efficient for the setting of small sizes. The reason is that the new formulation presented in this paper has a comparable convergence rate with the method in [13] for such cases, and much lower per-iteration complexity. When the problem size becomes large, the new approximation requires far more iterations to stabilize and thus results in a larger overall runtime. On the other hand, the average runtime of solving the GCP (10) is always larger than that of the method in [13, eq. (13)], since two approaches yields the same convergence speed while using the SOCP approximation significantly reduces the per-iteration complexity (cf. Fig. 2).

B. Convergence Results of Proposed Distributed Methods

In the second set of numerical experiments we explore the convergence properties of the proposed distributed methods, i.e., Algorithms 1 and 2. First the effect of the penalty parameter c on the convergence behavior of Algorithms 1 and 2 is investigated. To numerically verify Lemmas 1 and 2, we plot in Fig. 4(a) the convergence duality gap of Algorithms 1 and 2 with different values of c for a channel realization. The duality gap measures the distance between the values of the



(a) Duality gap with different values of c.



(b) Duality gap with $c_1 \neq c_2$ (see (33)).

Fig. 4. The duality gap of Algorithms 1 and 2 by 20000 iterations with $P_b = 35$ dBm , $P_{\text{dyn}} = 40$ dbm, $K = 2$ and different values of c for one channel realization.

augmented Lagrangian at iteration l and the one at the optimal solution of the ADMM procedure. More explicitly, for Algorithm 1 the duality gap is defined for the inner ADMM loop of the SCA$_n$ problem which is given by $\mathcal{L}^n(\boldsymbol{\vartheta}^l, \mathbf{s}^l, \boldsymbol{\zeta}^l, \boldsymbol{\xi}^l) - \mathcal{L}^n(\boldsymbol{\vartheta}^*, \mathbf{s}^*, \boldsymbol{\zeta}^*, \boldsymbol{\xi}^*)$ where $(\boldsymbol{\vartheta}^*, \mathbf{s}^*, \boldsymbol{\zeta}^*, \boldsymbol{\xi}^*)$ is the convergence point of the inner procedure. For Algorithm 2, the duality gap is defined as $\mathcal{L}(\boldsymbol{\vartheta}^l, \mathbf{s}^l, \boldsymbol{\zeta}^l, \boldsymbol{\xi}^l) - \mathcal{L}(\boldsymbol{\vartheta}^*, \mathbf{s}^*, \boldsymbol{\zeta}^*, \boldsymbol{\xi}^*)$ where $(\boldsymbol{\vartheta}^*, \mathbf{s}^*, \boldsymbol{\zeta}^*, \boldsymbol{\xi}^*)$ is the output of the algorithm. We also note that for Algorithm 1, Figs. 4(a) and 4(b) only demonstrate the convergence of the augmented Lagrangian of the ADMM inner loop for the SCA$_1$ problem, i.e., the first SCA iteration. As we can see, the convergence of both distributed algorithms improves as c decreases, and monotonic decrease in the augmented Lagrangian is guaranteed for large c, while this does not hold for small c. This observation is consistent with the analytical results in Lemmas 1 and 2. In Fig. 4(b) we illustrate how the convergence of Algorithms 1 and 2 improves by using different penalty parameters c_1 and c_2 in (33). As can be observed, the setup $c_1 = 20$, $c_2 = 0.02$ significantly reduces the number of iterations needed to converge since it

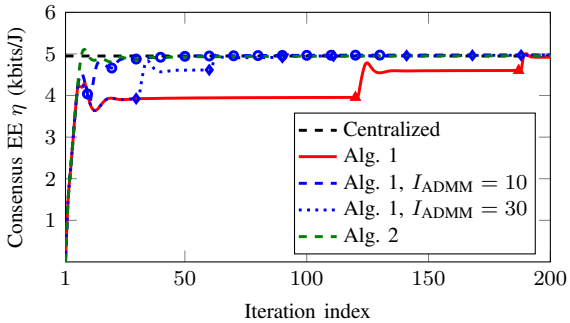


Fig. 5. Convergence behavior of Algorithms 1 and 2 with $P_b = 35$ dBm, $P_{\text{dp}} = 40$ dBm and $K = 2$. The markers denote the iterations where the SCA parameters are updated.

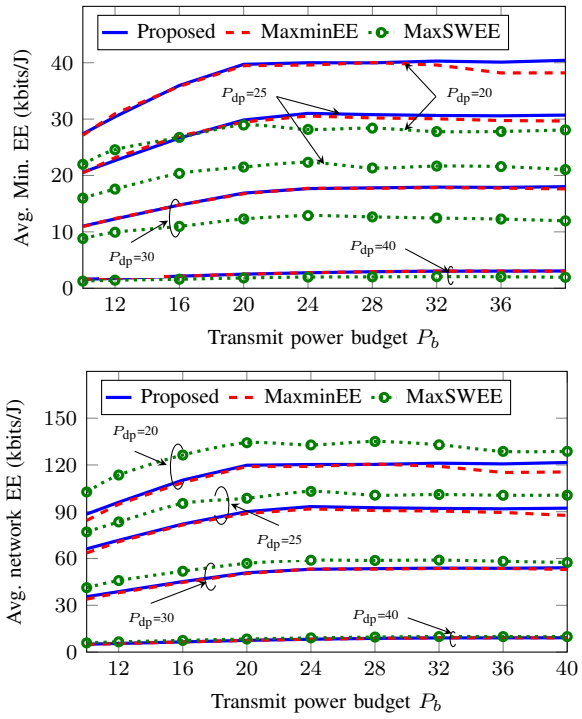


Fig. 6. Average EE versus the transmit power budget P_b for fixed value of total circuit power P_0 with $K = 2$.

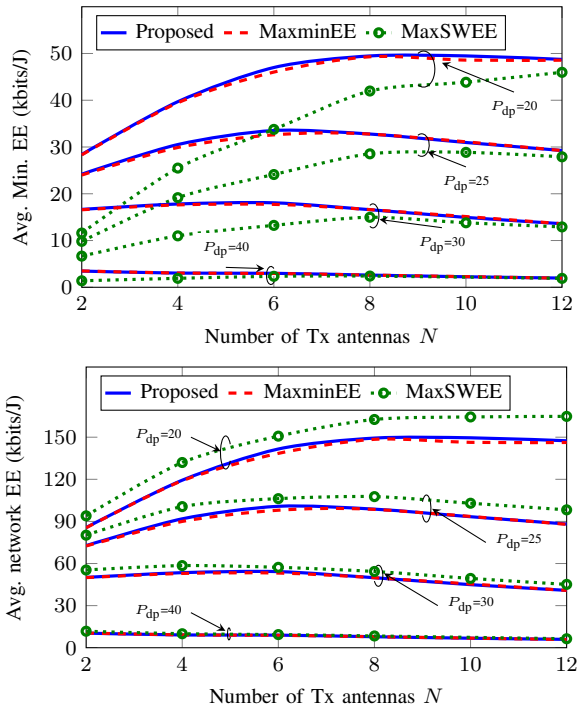


Fig. 7. Average EE versus the number of Tx antennas N with $P_b = 35$ dBm and $K = 2$.

successfully compensates for the imbalance among the penalty terms. We also see that Algorithm 2 achieves a faster convergence speed compared to Algorithm 1. For the remaining figures we set $c_1 = 20$ and $c_2 = 0.02$, which offers good convergence rate.

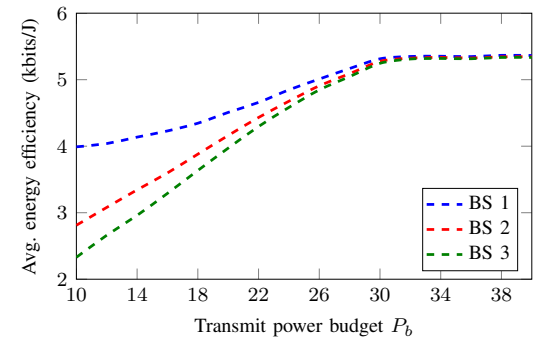
In Fig. 5 we compare the convergence of the consensus EE η of Algorithm 1, Algorithm 1 with a fixed number of ADMM iterations, and Algorithm 2 for one channel realization. We can see that the proposed decentralized algorithms obtain the same optimal solution as the centralized approach. Since the SCA parameters of Algorithm 1 are updated after the ADMM inner loop has completely converged, it usually requires a large number of iterations to output a solution. On the other hand, limiting the maximum number of iterations of the ADMM and updating the SCA parameters earlier, can significantly accelerate the convergence rate of Algorithm 1.

C. Achievable max-min EE

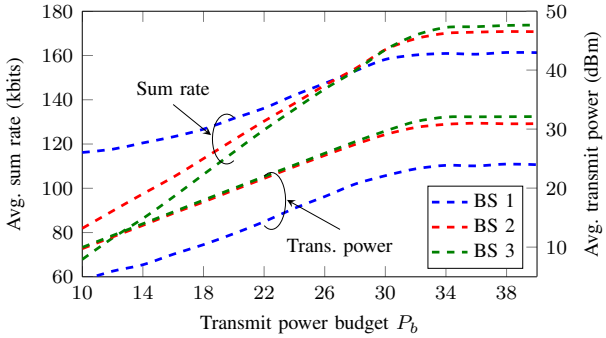
In the final set of numerical experiments, we illustrate the achieved EE versus different settings of the transmit power budget P_b, the number of transmit antennas N, and the total circuit power consumption P_0. For comparison purpose, we provide the performances of two existing EE schemes, the max-min EE in [10] and sum weighted EE maximization (maxSWEE) in [7]. Note that the maxSWEE scheme is given by $\max_{\mathbf{w}}\{\sum_{b=1}^{B} \alpha_b f_b(\mathbf{w})\}$ where α_b is the weighting factor, which is set to $\alpha_b = 1$ in this paper. The achieved EE is compared in terms of the achieved minimum EE (i.e., $\min_{b\in B}\{f_b(\mathbf{w}^*)\}$) and total network EE (i.e., $\sum_{b=1}^{B} f_b(\mathbf{w}^*)$). Each point of the curves in Figs. 6–8 is obtained by averaging from 1000 random channel realizations.

Fig. 6 plots the average achieved EE versus the transmit power budget for a fixed total circuit power consumption. We can see that, as the transmit power budget increases, the achieved EE first increases (when P_b is small) and then it remains constant (when P_b exceeds a certain level). This observation can be explained based on the logarithm function of data rate as follows. In the low transmit power regime, a small increase of transmit power could result in a significant increase of data rate leading to an increase in EE. On the other hand, when transmit power is large enough, further increasing the transmit power could reduce the EE performance because the data rate slightly increases, and thus the EE strategy does not use all transmit power budget. We also observe that the proposed max-min EE scheme achieves the same performance with the one in [10]. In addition, the max-min algorithm in general outperforms the maxSWEE scheme in terms of the minimum EE while it suffers a certain loss on the overall network EE compared to the maxSWEE approach. This is understood since the max-min EE scheme aims to improve the minimum-EE BS, not the total achieved EE.

Fig. 7 shows the average achieved EE as a function of the number of transmit antennas when the transmit power budget is fixed. As can be seen, the achieved EE increases with N when N is small and vice versa when N is large enough. We explain this result based on the two following facts. First, the data rate is a logarithmic function with N [33]. Second, the total consumed power linearly increases with N. That is to say, when N is large enough, the spatial gain cannot compensate for the additional circuit power, leading to a decrease in the achieved EE. As N becomes large, antenna selection has proved to be an efficient way to improve the EE



(a) Average EE versus P_b



(b) Average sum rate and transmit power versus P_b

Fig. 8. Achieved EE, sum rate and transmit power versus the transmit power budget P_b with $P_{\text{dp}} = 40$ dBm and $K = 2$.

as shown in [8].

Fig. 8 illustrates the average achieved sum rate and the average transmit power of an individual BS versus the transmit power budget with a fixed value of P_0. In particular we demonstrate how the EE of individual BSs varies in a more severe interference setting. For this purpose, users served by BSs 1, 2 and 3 are dropped to the cells within the distances of [50–200 m], [200–400 m] and [400–500 m] from their own serving BS. That is, the users served by BS 1 lie in their serving cell center, while those served by BSs 2 and 3 lie towards the cell edge. The sum rate and transmit power of each BS are calculated as $\sum_{k=1}^{K} r_{b_k}(\mathbf{w}^*)$ and $\sum_{k=1}^{K} \|\mathbf{w}_{b,k}^*\|^2$ $\forall b \in B$, respectively. Accordingly, the interference seen by users in BS 1 is smaller than that by users in BSs 2 and 3. As seen in Fig. 8(a), the EEs of three BSs are different if the transmit power budget P_b is small, i.e., below 32 dBm in this considered simulation setting. Due to the specific interference situation explained above, BSs 2 and 3 use more power but achieve lower sum rates in return, which is shown in Fig. 8(b). This explains the largest EE of BS 1. We can also see that for small P_b, the max-min EE fairness optimization in fact prevents BS 1 from transmitting at full power to establish fairness. On the other hand, the max-min fairness is achieved when P_b is large enough, i.e., when BSs 2 and 3 have sufficient power budget to deal with the path loss. Recall the results discussed above that, due to the EE optimal design, the transmit power and the achieved sum rate of all BSs will remain unchanged when P_b exceeds a certain threshold.

VII. Conclusion

This paper has studied the distributed solutions for the problem of the EE fairness optimization for multicell multiuser MISO downlink systems, in which minimum EE among all BSs is maximized. To this end, we have transformed the original nonconvex design problem into a tractable form where the hidden convexity is more exposed. Exploiting the hidden convexity, we have developed two algorithms based on the combination of SCA and ADMM, which are suitable for systems with limited backhaul signaling. In the first algorithm, the remaining nonconvexity of the problem was replaced by the convex approximations using SCA principle. Then we have introduced the sets of local and global variables to translate the approximate problem into the ADMM applicable form, before applying the ADMM to solve the arrived convex programs at each iteration of the SCA procedure. More specifically, in one iteration of the SCA loop, we carried out the ADMM update by allowing the BSs to exchange the required information until those variables converge to optimal consensus values. Numerical results have demonstrated that, in some cases, Algorithm 1 requires to exchange a large amount of information among all BSs. Therefore, we have proposed the second algorithm for which the backhaul overhead is remarkably reduced. In the latter algorithm, instead of using the ADMM to find the optimal solution of the SCA, we have applied the ADMM to guide the local and global variables towards consensus values while the SCA was used to update the local variables. The convergence analysis of the proposed algorithms has been provided and discussed. Simulation results have been provided to evaluate the effectiveness of the proposed algorithms.

Appendix A
Proof of Lemma 1

The convergence analysis of Algorithm 1 can be done exactly in the same way as in [29]. Herein, we provide another convergence proof for Algorithm 1 which is also the key to establishing the convergence results of Algorithm 2. In particular our new proof is inspired by the recent work of [31] in which the authors show monotonicity of the Lagrangian function by a proper choice of the penalty parameter. By the power constraint at each BS we can assume that $\mathcal{S}_b^{n+1}$ given in (17) is a nonempty closed convex set. To proceed let us consider the problem SCA_{n+1} and denote by $I_{\mathcal{S}_b^{n+1}}$ the indicator function of $\mathcal{S}_b^{n+1}$, i.e., $I_{\mathcal{S}_b^{n+1}}(\mathbf{x}) = 0$ if $\mathbf{x} \in \mathcal{S}_b$ and $+\infty$ otherwise. Then the constrained optimization problem in (25) can be rewritten as

$$\begin{aligned}\mathbf{s}_b^{l+1} &= \arg\min_{\mathbf{s}_b \in \mathcal{S}_b^{n+1}} \; -\eta_b + \xi_b^l(\eta_b - \eta^{l+1}) + (\boldsymbol{\zeta}_b^l)^T(\boldsymbol{\theta}_b - \boldsymbol{\nu}_b^{l+1}) \\ &\qquad + \frac{c}{2}\left((\eta_b - \eta^{l+1})^2 + \|\boldsymbol{\theta}_b - \boldsymbol{\nu}_b^{l+1}\|_2^2\right) + I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b) \\ &\triangleq \arg\min_{\mathbf{s}_b \in \mathcal{S}_b^{n+1}} \mathcal{L}_b^{l+1}(\mathbf{s}_b) + I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b).\end{aligned} \tag{34}$$

It is known that if $\mathbf{s}_b^*$ is the solution of (34) then $0 \in \nabla_{\mathbf{s}_b}\mathcal{L}_b^{l+1}(\mathbf{s}_b^*) + \partial I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b^*)$, where the notation $\partial I_{\mathcal{S}_b^{n+1}}(\mathbf{x})$ denotes the set of all subgradients of $I_{\mathcal{S}_b^{n+1}}(\mathbf{x})$ [34]. That is $\partial I_{\mathcal{S}_b^{n+1}}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R} | \langle \mathbf{u}, \mathbf{y} - \mathbf{x}\rangle \le 0 \; \forall \mathbf{y} \in \mathcal{S}_b^{n+1}\}$. Note that

$\partial I_{\mathcal{S}_b^{n+1}}(\mathbf{x})$ is in fact the normal cone of $\mathcal{S}_b^{n+1}$ at $\mathbf{x} \in \mathcal{S}_b^{n+1}$ [34, pp. 215]. Now we are ready to show monotonic decrease of the augmented Lagrangian between two consecutive iterations in the inner loop of Algorithm 1. Let us split the LHS of (31) into three terms as

$$\begin{aligned}
&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l+1}, \boldsymbol{\xi}^{l+1}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&= [\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l+1}, \boldsymbol{\xi}^{l+1}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})] \\
&\quad + [\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})] \\
&\quad + [\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})].
\end{aligned} \tag{35}$$

The first term in (35) refers to the Lagrange multipliers update which can be expressed as

$$\begin{aligned}
&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l+1}, \boldsymbol{\xi}^{l+1}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&= \frac{1}{c}\sum_{b=1}^{B}((\xi_b^{l+1} - \xi_b^{l})^2 + \|\boldsymbol{\zeta}_b^{l+1} - \boldsymbol{\zeta}_b^{l}\|_2^2).
\end{aligned}$$

By the step of updating local variable η_b, we obtain the following optimality condition

$$1 - \xi_b^{l} - c(\eta_b^{l+1} - \eta^{l+1}) = 1 - \xi_b^{l+1} \in \partial I_{\mathcal{S}_b^{n+1}}(\eta_b^{l+1}). \tag{36}$$

Since all constraints in $\mathcal{S}_b^{n+1}$ are continuously differentiable, it is known that $I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b)$ is Lipschitz continuous. Let d be the Lipschitz constant of $I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b)$. Then $\|g\|_2 \leq d$ for all $g \in \partial I_{\mathcal{S}_b^{n+1}}(\eta_b^{l+1})$. As a result we have $\sum_{b=1}^{B}(\xi_b^{l+1} - \xi_b^{l})^2 \leq \sum_{b=1}^{B} d^2(\eta_b^{l+1} - \eta_b^{l})^2$. Similarly, the above arguments can be applied to achieve $\sum_{b=1}^{B}\|\boldsymbol{\zeta}_b^{l+1} - \boldsymbol{\zeta}_b^{l}\|_2^2 \leq \sum_{b=1}^{B} d^2\|\boldsymbol{\theta}_b^{l+1} - \boldsymbol{\theta}_b^{l}\|_2^2$. Thus, the first term can be bounded as

$$\begin{aligned}
&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l+1}, \boldsymbol{\xi}^{l+1}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&\leq \sum_{b=1}^{B}\frac{d^2}{c}((\eta_b^{l+1} - \eta_b^{l})^2 + \|\boldsymbol{\theta}_b^{l+1} - \boldsymbol{\theta}_b^{l}\|_2^2).
\end{aligned} \tag{37}$$

For the second term in (35) we notice that $\mathbf{s}_b^{l+1}$ is the solution to (34) and thus the following equality holds

$$\begin{aligned}
&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&= \sum_{b=1}^{B}(\mathcal{L}_b^{l+1}(\mathbf{s}_b^{l+1}) - \mathcal{L}_b^{l+1}(\mathbf{s}_b^{l})) \\
&= -\sum_{b=1}^{B}\frac{c}{2}((\eta_b^{l} - \eta_b^{l+1})^2 + \|\boldsymbol{\theta}_b^{l} - \boldsymbol{\theta}_b^{l+1}\|_2^2) \\
&\quad + \sum_{b=1}^{B}\left\langle \partial I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b^{l+1}), (\mathbf{s}_b^{l} - \mathbf{s}_b^{l+1})\right\rangle \\
&\quad - \sum_{b=1}^{B}\left\langle (\nabla_{\boldsymbol{s}_b}\mathcal{L}_b(\mathbf{s}_b^{l+1}) + \partial I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b^{l+1})), (\mathbf{s}_b^{l} - \mathbf{s}_b^{l+1})\right\rangle \\
&= -\sum_{b=1}^{B}\frac{c}{2}((\eta_b^{l} - \eta_b^{l+1})^2 + \|\boldsymbol{\theta}_b^{l} - \boldsymbol{\theta}_b^{l+1}\|_2^2) \\
&\quad + \sum_{b=1}^{B}\left\langle \partial I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b^{l+1}), (\mathbf{s}_b^{l} - \mathbf{s}_b^{l+1})\right\rangle.
\end{aligned}$$

Note that $\left\langle \partial I_{\mathcal{S}_b^{n+1}}(\mathbf{s}_b^{l+1}), (\mathbf{s}_b^{l} - \mathbf{s}_b^{l+1})\right\rangle \leq 0$ due to the definition of $\partial I_{\mathcal{S}_b^{n+1}}(\mathbf{x})$. Thus, the second term can be bounded by

$$\begin{aligned}
&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&\leq -\sum_{b=1}^{B}\frac{c}{2}((\eta_b^{l} - \eta_b^{l+1})^2 + \|\boldsymbol{\theta}_b^{l} - \boldsymbol{\theta}_b^{l+1}\|_2^2).
\end{aligned} \tag{38}$$

Finally, the third term in (35) is equivalent to

$$\begin{aligned}
&\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&= -\left\langle \nabla_{\boldsymbol{\vartheta}}\mathcal{L}^{n+1}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}), \boldsymbol{\vartheta}^{l} - \boldsymbol{\vartheta}^{l+1}\right\rangle \\
&\quad - \sum_{b=1}^{B}\frac{c}{2}((\eta^{l} - \eta^{l+1})^2 + \|\boldsymbol{\nu}_b^{l+1} - \boldsymbol{\nu}_b^{l}\|_2^2) \\
&= -\sum_{b=1}^{B}\frac{c}{2}((\eta^{l+1} - \eta^{l})^2 + \|\boldsymbol{\nu}_b^{l+1} - \boldsymbol{\nu}_b^{l}\|_2^2),
\end{aligned} \tag{39}$$

since $\boldsymbol{\vartheta}^{l+1}$ is the optimal solution of the unconstrained optimization problem $\min_{\boldsymbol{\vartheta}} \{\mathcal{L}^{n+1}(\boldsymbol{\vartheta}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})\}$. Combining (37), (38), and (39) we obtain (31) which completes the proof.

Appendix B
Proof of Lemma 2

The proof of Lemma 2 follows the same steps as those Lemma 1, but some modifications are made since the SCA parameters in Algorithm 2 are immediately updated right after every ADMM iteration. First we rewrite the LHS of (32) as

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l+1}, \boldsymbol{\xi}^{l+1}) - \mathcal{L}(\boldsymbol{\vartheta}^{l}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) \\
&= [\mathcal{L}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l+1}, \boldsymbol{\xi}^{l+1}) - \mathcal{L}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})] \\
&\quad + [\mathcal{L}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l+1}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})] \\
&\quad + [\mathcal{L}(\boldsymbol{\vartheta}^{l+1}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l}) - \mathcal{L}(\boldsymbol{\vartheta}^{l}, \mathbf{s}^{l}, \boldsymbol{\zeta}^{l}, \boldsymbol{\xi}^{l})].
\end{aligned} \tag{40}$$

We note that the first and the last terms can be bounded exactly as done in (37) and (39), respectively. Thus to prove Lemma 2 we only need to deal with the second term in (40). Towards this end let us introduce the feasible set $\tilde{\mathcal{S}}_b \triangleq \{\mathbf{s}_b \mid (7c), (7f), (7g), (16b), (16c)\}$ and again $I_{\tilde{\mathcal{S}}_b}(\mathbf{s}_b)$ denotes the indicator function of $\tilde{\mathcal{S}}_b$. The update of local variable step is now equivalent to the following convex program

$$\begin{aligned}
\min_{\mathbf{s}_b \in \tilde{\mathcal{S}}_b} \quad & \mathcal{L}_b^{l+1}(\mathbf{s}_b) + I_{\tilde{\mathcal{S}}_b}(\mathbf{s}_b) && \text{(41a)} \\
\text{subject to} \quad & u_b^{l+1}(\mathbf{s}_b) \leq 0 \quad b \in \mathcal{B}, && \text{(41b)} \\
& v_{b_k}^{l+1}(\mathbf{s}_b) \leq 0, \quad b \in \mathcal{B}, && \text{(41c)}
\end{aligned}$$

where $\mathcal{L}_b^{l+1}(\mathbf{s}_b)$ is given in (34), $u_b^{l+1}(\mathbf{s}_b) = \eta_b - \frac{2z_b^l}{t_b^l}z_b + \frac{(z_b^l)^2}{(t_b^l)^2}t_b$, and $v_{b_k}^{l+1}(\mathbf{s}_b) = g_{b_k} - \frac{2\Re(\mathbf{h}_{b,b_k}^l \mathbf{w}_{b_k})}{q_{b_k}^l} + \frac{|\mathbf{h}_{b,b_k}\mathbf{w}_{b_k}^l|^2 q_{b_k}}{(q_{b_k}^l)^2}$. We remark that (41) is considered instead of (34) since $u_b^{l+1}(\mathbf{s}_b)$ and $v_{b_k}^{l+1}(\mathbf{s}_b)$ are replaced by new constraints after each iteration. Let $\pi_b \in \mathbb{R}$ and $\omega_{b_k} \in \mathbb{R}$ be the Lagrangian

multipliers of (41b) and (41c), respectively. Then we can rewrite the second term in (40) as

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{\vartheta}^{l+1},\mathbf{s}^{l+1},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})-\mathcal{L}(\boldsymbol{\vartheta}^{l+1},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})\\
&=\sum_{b=1}^{B}(\mathcal{L}_b^{l+1}(\mathbf{s}_b^{l+1})-\mathcal{L}_b^{l+1}(\mathbf{s}_b^{l}))\\
&=-\sum_{b=1}^{B}\frac{c}{2}((\eta_b^l-\eta_b^{l+1})^2+\|\boldsymbol{\theta}_b^l-\boldsymbol{\theta}_b^{l+1}\|_2^2)\\
&\quad+\sum_{b=1}^{B}\Big[\Big\langle((\partial I_{\tilde{\mathcal{S}}_b}(\mathbf{s}_b^{l+1})+\pi_b\nabla_{\mathbf{s}_b}u_b^{l+1}(\mathbf{s}_b^{l+1})\\
&\qquad+\textstyle\sum_{k=1}^{K}\omega_{b_k}\nabla_{\mathbf{s}_b}v_{b_k}^{l+1}(\mathbf{s}_b^{l+1})),(\mathbf{s}_b^{l}-\mathbf{s}_b^{l+1})\Big\rangle\Big]\\
&\quad-\sum_{b=1}^{B}\Big[\Big\langle((\nabla_{\boldsymbol{s}_b}\mathcal{L}_b^{l+1}(\mathbf{s}_b^{l+1})+\partial I_{\tilde{\mathcal{S}}_b}(\mathbf{s}_b^{l+1})+\pi_b\nabla_{\mathbf{s}_b}u_b^{l+1}(\mathbf{s}_b^{l+1})\\
&\qquad+\textstyle\sum_{k=1}^{K}\omega_{b_k}\nabla_{\mathbf{s}_b}v_{b_k}^{l+1}(\mathbf{s}_b^{l+1})),(\mathbf{s}_b^{l}-\mathbf{s}_b^{l+1})\Big\rangle\Big].
\end{aligned}\tag{42}$$

Since $\mathbf{s}_b^{l+1}$ is optimal to (41), the KKT conditions result in

$$\begin{cases}
0\in\big(\nabla_{\mathbf{s}_b}\mathcal{L}_b^{l+1}(\mathbf{s}_b^{l+1})+\pi_b\nabla_{\mathbf{s}_b}u_b^{l+1}(\mathbf{s}_b^{l+1})\\
\qquad+\sum_{k=1}^{K}\omega_{b_k}\nabla_{\mathbf{s}_b}v_{b_k}^{l+1}(\mathbf{s}_b^{l+1})+\partial I_{\tilde{\mathcal{S}}_b}(\mathbf{s}_b^{l+1})\big)\\
\pi_b\geq 0,\omega_{b_k}\geq 0,\ \pi_b u_b^{l+1}(\mathbf{s}_b^{l+1})=0,\ \omega_{b_k}v_{b_k}^{l+1}(\mathbf{s}_b^{l+1})=0.
\end{cases}\tag{43}$$

Due to the linearity of $u_b^{l+1}(\mathbf{s}_b)$, the following equality holds

$$\begin{aligned}
&u_b^{l+1}(\mathbf{s}_b^{l+1})-u_b^{l+1}(\mathbf{s}_b^{l})+\big\langle\nabla_{\mathbf{s}_b}u_b^{l+1}(\mathbf{s}_b^{l+1}),(\mathbf{s}_b^{l}-\mathbf{s}_b^{l+1})\big\rangle\\
&=0\\
\Leftrightarrow\ &u_b^{l+1}(\mathbf{s}_b^{l+1})-u_b^{l}(\mathbf{s}_b^{l})\ +\big\langle\nabla_{\mathbf{s}_b}u_b^{l+1}(\mathbf{s}^{l+1}),(\mathbf{s}_b^{l}-\mathbf{s}_b^{l+1})\big\rangle\\
&=-\delta_b^l(\mathbf{z},\mathbf{t}),
\end{aligned}\tag{44}$$

where

$$\begin{aligned}
\delta_b^l(\mathbf{z},\mathbf{t})&\triangleq u_b^l(\mathbf{s}_b^l)-u_b^{l+1}(\mathbf{s}_b^l)\\
&=-\frac{2z_b^{l-1}}{t_b^{l-1}}z_b^l-\frac{(z_b^{l-1})^2}{(t_b^{l-1})^2}t_b^l+\frac{2z_b^l}{t_b^l}z_b^l-\frac{(z_b^l)^2}{(t_b^l)^2}t_b^l\\
&=2z_b^l(\frac{z_b^l}{t_b^l}-\frac{z_b^{l-1}}{t_b^{l-1}})-t_b^l(\frac{(z_b^l)^2}{(t_b^l)^2}-\frac{(z_b^{l-1})^2}{(t_b^{l-1})^2})\\
&=t_b^l(\frac{z_b^l}{t_b^l}-\frac{z_b^{l-1}}{t_b^{l-1}})^2.
\end{aligned}$$

In the same way we have

$$v_{b_k}^{l+1}(\mathbf{s}_b^{l+1})-v_{b_k}^{l}(\mathbf{s}_b^{l})+\big\langle\nabla_{\mathbf{s}}v_{b_k}^{l+1}(\mathbf{s}^{l+1}),(\mathbf{s}^{l}-\mathbf{s}^{l+1})\big\rangle=-\delta_{b_k}^l(\mathbf{w},\mathbf{q})\tag{45}$$

where $\delta_{b_k}^l(\mathbf{w},\mathbf{q})\triangleq q_{b_k}^l(\frac{\Re(\mathbf{h}_{b,b_k}^l\mathbf{w}_{b,b_k}^l)}{q_{b_k}^l}-\frac{\Re(\mathbf{h}_{b,b_k}^{l-1}\mathbf{w}_{b,b_k}^{l-1})}{q_{b_k}^{l-1}})^2$.

Thus, by (43), (44), (45) and noting that $\pi_b(u_b^{l+1}(\mathbf{s}_b^{l+1})-u_b^l(\mathbf{s}_b^l))\geq 0$, $\omega_{b_k}(v_{b_k}^{l+1}(\mathbf{s}_b^{l+1})-v_{b_k}^l(\mathbf{s}_b^l))\geq 0$, and $\big\langle\partial I_{\tilde{\mathcal{S}}_b}(\mathbf{s}_b^{l+1}),(\mathbf{s}_b^l-\mathbf{s}_b^{l+1})\big\rangle\leq 0$, we obtain an upper bound of (42) as

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{\vartheta}^{l+1},\mathbf{s}^{l+1},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})-\mathcal{L}(\boldsymbol{\vartheta}^{l+1},\mathbf{s}^{l},\boldsymbol{\zeta}^{l},\boldsymbol{\xi}^{l})\\
&\leq-\sum_{b=1}^{B}\frac{c}{2}((\eta_b^l-\eta_b^{l+1})^2+\|\boldsymbol{\theta}_b^l-\boldsymbol{\theta}_b^{l+1}\|_2^2)-\sum_{b=1}^{B}\delta_b^l,
\end{aligned}\tag{46}$$

where $\delta_b^l\triangleq\pi_b\delta_b^l(\mathbf{z},\mathbf{t})+\sum_{k=1}^{K}\omega_{b_k}\delta_{b_k}^l(\mathbf{w},\mathbf{q})$. Hence, combining (37), (39), and (46) we have (32) and thus completes the proof.

REFERENCES

[1] K. G. Nguyen, L. N. Tran, Q. D. Vu, and M. Juntti, "Distributed energy efficiency fairness optimization by ADMM in multicell MISO downlink," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kualalumpur, Malaysia, May 2016, pp. 1–6.

[2] D. Feng, C. Jiang, G. Lim, L. J. Cimini, G. Feng, and G. Y. Li, "A survey of energy-efficient wireless communication," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 167–178, Feb. 2013.

[3] B. B. Haro, S. Zazo, and D. P. Palomar, "Energy efficient collaborative beamforming in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 496 – 510, Jan. 2014.

[4] G. Miao, N. Himayat, and G. Y. Li, "Energy-efficient link adaptation in frequency-selective channels," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 545 – 554, Feb. 2010.

[5] E. Björnson, L. Sanguinetti, J. Hoydis, and M. Debbah, "Optimal design of energy-efficient multi-user MIMO systems: Is massive MIMO the answer?" *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3059 – 3075, Jun. 2015.

[6] Z. Xu, C. Yang, G. Y. Li, Y. Liu, and S. Xu, "Energy-efficient CoMP precoding in heterogeneous networks," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 1005 – 1017, Feb. 2014.

[7] S. He, Y. Huang, L. Yang, and B. Ottersten, "Coordinated multicell multiuser precoding for maximizing weighted sum energy efficiency," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 741–751, Feb. 2014.

[8] O. Tervo, L.-N. Tran, and M. Juntti, "Optimal energy-efficient transmit beamforming for multi-user MISO downlink," *IEEE Trans. Signal Process.*, vol. 63, no. 20, pp. 5574 – 5588, Oct. 2015.

[9] D. Nguyen, L.-N. Tran, P. Pirinen, and M. Latva-aho, "Precoding for full duplex multiuser MIMO systems: Spectral and energy efficiency maximization," *IEEE Trans. Signal Process.*, vol. 61, no. 16, pp. 4038–4050, Aug. 2013.

[10] S. He, Y. Huang, S. Jin, F. Yu, and L. Yang, "Max-min energy efficient beamforming for multicell multiuser joint transmission systems," *IEEE Commun. Lett.*, vol. 17, no. 10, pp. 1956–1959, Oct. 2013.

[11] B. Du, C. Pan, W. Zhang, and M. Chen, "Distributed energy-efficient power optimization for CoMP systems with max-min fairness," *IEEE Commun. Lett.*, vol. 18, no. 6, pp. 999–1002, Jun. 2014.

[12] A. Zappone, L. Sanguinetti, G. Bacci, E. Jorswieck, and M. Debbah, "Energy-efficient power control: A look at 5G wireless technologies," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1668–1683, Apr. 2016.

[13] K.-G. Nguyen, L.-N. Tran, O. Tervo, Q.-D. Vu, and M. Juntti, "Achieving energy efficiency fairness in multicell multiuser MISO downlink," *IEEE Commun. Lett.*, vol. 19, no. 8, pp. 1426 – 1429, Aug. 2015.

[14] A. Zappone and E. Jorswieck, "Energy efficiency in wireless networks via fractional programming theory," *Foundations and Trends in Communications and Information Theory*, vol. 11, no. 3-4, pp. 185–396, 2015.

[15] C. Shen, T.-H. Chang, K.-Y. Wang, Z. Qiu, and C.-Y. Chi, "Distributed robust multicell coordinated beamforming with imperfect CSI: An ADMM approach," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2988–3003, Jun. 2012.

[16] Q.-D. Vu, L.-N. Tran, M. Juntti, and E.-K. Hong, "Energy-efficient bandwidth and power allocation for multi-homing networks," *IEEE Trans. Signal Process.*, vol. 63, no. 7, pp. 1684–1699, Apr. 2015.

[17] O. Arnold, F. Richter, G. Fettweis, and O. Blume, "Power consumption modeling of different base station types in heterogeneous cellular networks," in *IEEE Future Network & Mobile Summit 2010*, Florence, Italy, 2010, pp. 1–8.

[18] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. A. Imran, D. Sabella, M. J. Gonzalez, O. Blume, and A. Fehske, "How much energy is needed to run a wireless network?" *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 40–49, Oct. 2011.

[19] Y.-F. Liu, Y.-H. Dai, and Z.-Q. Luo, "Coordinated beamforming for MISO interference channel: Complexity analysis and efficient algorithms," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1142–1157, Mar. 2011.

[20] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 57–73, Feb. 2008.

[21] J. Crouzeix and J. Ferland, "Algorithms for generalized fractional program," *Mathematical Programming*, vol. 52, pp. 191–207, May 1991.

[22] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Operations Research*, vol. 26, no. 4, pp. 681–683, Jul.-Aug. 1978.

[23] I. MOSEK ApS, 2014, [Online]. Available: www.mosek.com.

[24] Gurobi Optimization, Gurobi Optimizer Version 5.1., Mar./Apr. 2013. [Online]. Available: http://www.gurobi.com/.

[25] Ben-Tal, Ahron, Nemirovski, and Arkadi, *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*. MPS-SIAM Series on Optimization, SIAM, 2001.

[26] E. Björnson, R. Zakhour, D. Gesbert, and B. Ottersten, "Cooperative multicell precoding: Rate region characterization and distributed strategies with instantaneous and statistical CSI," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4298–4310, Aug 2010.

[27] R. Mochaourab and E. A. Jorswieck, "Optimal beamforming in interference networks with perfect local channel information," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1128–1141, March 2011.

[28] Y. Huang, G. Zheng, M. Bengtsson, K. K. Wong, L. Yang, and B. Ottersten, "Distributed multicell beamforming with limited intercell coordination," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 728–738, Feb 2011.

[29] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.,*, vol. 3, no. 1, pp. 1–122, 2011.

[30] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.

[31] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 337–364, 2016.

[32] R. T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM J. Control Optim. (SICON)*, vol. 14, no. 5, pp. 877–898, Aug 1976.

[33] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 742–758, Oct 2014.

[34] R. T. Rockafellar, *Convex Analysis*. Princeton University Press, 1970.